AD-A037 059 OPERATIONS RESEARCH INC SILVER SPRING MD F/G 15/7
DEVELOPMENT OF THE FLEET AIR READINESS TRAINING MODEL FOR THE I--ETC(U)
MAR 71 T N KYLE, R J CRAIG, M C FISK N00025-67-C-0031
UNCLASSIFIED ORI-TR-648-VOL-4 NL

1 OF 1
AD A037059

END
DATE FILMED
4-77

ORI

Operations Research, Inc. A LEASCO Company

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Operations Research, Inc.	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP 4 - Excluded from General Declassification Schedule

3. REPORT TITLE

Development of the Fleet Air Readiness Training Model for the Integrated Facilities Requirements Study (IFRS) Phase III, Volume IV. FRT Programmer's Manual Readiness Training (FRT) Management Planning Tool.

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final Report, 31 March 1971

5. AUTHOR(S) (First name, middle initial, last name)

Thomas N. Kyle, W. Liggett
R. J. Craig, F. McCoy
M. C. Fisk, R. Messalle
R. Yockman

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
31 March 1971	94	N. A.

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N00025-67-0031 (NBy-78672) b. PROJECT NO. N. A. c. N00025-67-C-0031	ORI-TR-648-Vol-4 Vol IV of IV
	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) N. A.

10. DISTRIBUTION STATEMENT

Statement No. 1 -
Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
N. A.	Naval Facilities Engineering Command Department of the Navy Washington, DC

13. ABSTRACT

This report documents the Fleet Readiness Training (FRT) Planning model developed as part of the third phase of the Integrated Facilities Requirements Study (IFRS).

In Phase I, two analytic submodels were developed. The first, a Logistics Support Requirements Generator, estimates personnel, aircraft, and fuel requirements for each phase of undergraduate pilot training at the Naval Air Training Command (NATRACOM). The second, a Pacing Facilities Requirements submodel, calculates facility requirements for each phase of training.

The purpose of the Phase II study was to develop a preliminary total systems IFRS management planning tool (including the two submodels developed in Phase I, as well as Base Loading, Facilities Excess/Deficiency, and Total Cost submodels), and automate the model so that it provides quick, accurate, and relevant information for use in the decision-making process. This Static IFRS model has been in continuous operation since March 1970.

The purpose of the Phase III study was to refine the Static IFRS model and to expand the IFRS concept by developing three additional planning tools for use by Navy decision-makers as follows:

DD FORM 1 NOV 68 1473 (PAGE 1) 270 900 SB

S/N 0101-807-6801

Security Classification

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Facilities						
Requirements						
Dynamic						
Optimization						
Fleet						
Training						
Aircraft						
Pilot						
Simulation						
Programming						
Management						
Planning						
Static						
Air						
Readiness						
Model						

Item #13 (Abstract) continued

- Dynamic planning tool
- Optimization model
- Fleet Readiness Training Squadron planning tool.

The Dynamic planning tool simulates the undergraduate pilot training program on a weekly basis whereas the Static IFRS assumes an even annual flow of students. The Optimization model has two segments - a PTR Maximizer that calculates the maximum annual pilot training rate (PTR) possible for a given facilities inventory and a MCON Minimizer that calculates the minimum facility cost phase-to-base assignment for a desired PTR. The Fleet Readiness Training (FRT) model provides planning information for the readiness training squadrons and is designed similarly to the Static IFRS model. The Phase III documentation consists of the following four reports:

- The Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 645
- Development of the Automated Dynamic Model for the Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 646
- Development of the Optimization Model for the Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 647
- Development of the Fleet Air Readiness Training Model for the Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 648.

This report documents the Fleet Readiness Training (FRT) Model. Volume I contains a Summary of the FRT model and the functional relationships. Appendix B contains the present squadron planning factors and is under separate cover in Volume II. Volume III contains the User's Manual stating how to use the planning tool. The Programmer's Manual is contained in Volume IV.

OPERATIONS RESEARCH, Inc.

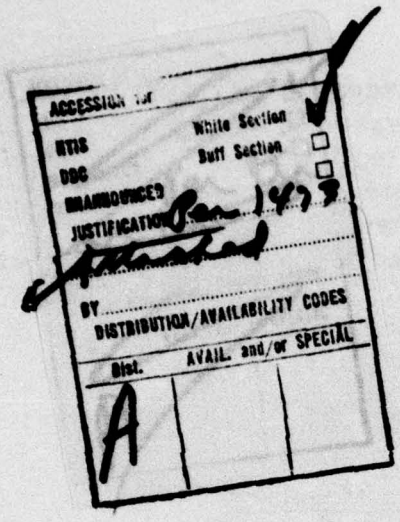
SILVER SPRING, MARYLAND

Development of the Fleet Air Readiness Training Model for the Integrated Facilities Requirements Study (IFRS) Phase III

Volume IV - FRT Programmer's Manual

31 March 1971

Prepared under Contract N00025-67-C-0031 (NBy-78672)
for the Naval Facilities Engineering Command
Department of the Navy
Washington, D.C.



FOREWORD

This report documents the Fleet Readiness Training (FRT) planning model developed as part of the third phase of the Integrated Facilities Requirements Study (IFRS). It has been prepared for the Systems Analysis Division of the Office of the Assistant Commander for Facilities Planning (Code 20), Naval Facilities Engineering Command (NAVFAC), Department of the Navy, as part of Contract N00025-67-C-0031 (NBy-78672) awarded to Operations Research, Inc., in June 1970.

In Phase I, two analytic submodels were developed. The first, a Logistics Support Requirements Generator, estimates personnel, aircraft, and fuel requirements for each phase of undergraduate pilot training at the Naval Air Training Command (NATRACOM). The second, a Pacing Facilities Requirements submodel, calculates facility requirements for each phase of training.

The purpose of the Phase II study was to develop a preliminary total systems IFRS management planning tool (including the two submodels developed in Phase I, as well as Base Loading, Facilities Excess/Deficiency, and Total Cost submodels), and automate the model so that it provides quick, accurate, and relevant information for use in the decision-making process. This Static IFRS model has been in continuous operation since March 1970.

The purpose of the Phase III study was to refine the Static IFRS model and to expand the IFRS concept by developing three additional planning tools for use by Navy decision-makers as follows:

- Dynamic planning tool
- Optimization model
- Fleet Readiness Training Squadron planning tool.

The Dynamic planning tool simulates the undergraduate pilot training program on a weekly basis whereas the Static IFRS assumes an even annual flow of students. The Optimization model has two segments—a PTR Maximizer that calculates the maximum annual pilot training rate (PTR) possible for a given facilities inventory and a MCON Minimizer that calculates the minimum facility cost phase-to-base-assignment for a desired PTR. The Fleet Readiness Training (FRT) model provides planning information for the readiness training squadrons and is designed similarly to the Static IFRS model. The Phase III documentation consists of the following four reports:

- The Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 645
- Development of the Automated Dynamic Model for the Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 646
- Development of the Optimization Model for the Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 647
- Development of the Fleet Air Readiness Training Model for the Integrated Facilities Requirements Study (IFRS) Phase III, ORI TR 648.

This report documents the Fleet Readiness Training (FRT) model. Volume I contains a summary of the FRT model and the functional relationships. Appendix B contains the present squadron planning factors and is under separate cover in Volume II. Volume III contains the User's Manual stating how to use the planning tool. The Programmer's Manual is contained in Volume IV.

These IFRS models were developed and programmed by the staff members of the Economic Analysis Division of Operations Research, Inc., under the direction of Dr. William J. Leininger, vice president and division director and Thomas N. Kyle, program director. The project team members included R. J. Craig, M. C. Fisk, W. Liggett, F. McCoy, R. Messalle, and R. Yockman.

Mr. Dennis Whang of the Systems Analysis Division of Facilities Planning was contract monitor for NAVFAC. In addition, valuable assistance was provided by many other Navy personnel including, in particular, those in the Office of the Staff Civil Engineer and the Training/Plans Division of the Naval Operations, and in the Systems Analysis Division of NAVFAC. The authors gratefully acknowledge the contributions made by all of these people to the development of the IFRS models.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. INTRODUCTION

1.1 This volume describes the overall system characteristics and flow for all computer programs included in the automated Fleet Readiness Training (FRT) model. The purpose of the programmer's manual is to provide the verbal description, flow charts, variable dictionary, program and subroutine dictionary, and program listing for each of the computer programs that constitute the automated model of the FRT. This programmer's manual provides Navy personnel with the information required to understand the logic of the programming and to make changes to the programs as necessary.

1.2 The FRT programs have been written in FORTRAN for use on a General Electric (GE) 635, Mark II, time-sharing computer system. It is assumed that the programmer using this documentation is fully acquainted with this system.[1/]

1.3 Because of the "in core" word limitation imposed by the GE 635 time-sharing computer, the FRT model is comprised of five different computer programs. This system will only permit a program of approximately 12,800 36-bit words to reside within the computer memory at any one time. Since the total

---

1/ Converting the computer programs contained in the automated FRT model for use on other FORTRAN IV systems would require major revisions to each individual computer program's input and output, due to a special feature in the GE time-sharing FORTRAN that allows unformatted input and output, an adaptation of a BASIC language feature. The authors of the FRT computer programs utilized this feature, when practical, to provide the user with maximum terminal input flexibility. In addition, storage restrictions might, in other FORTRAN IV systems, require resegmenting the FRT system so maximum program storage requirements would not be violated.

FRT model is much longer than this limit (approximately 21,000 36-bit words in the unsegmented version), it was necessary to use five operating programs. Table 1 lists each of these programs and their source and compiled names, lengths, and storage units. The asterisk in the sixth character "*" in the compiled name allows these programs to be accessed by all users (with certain restrictions) having similar GE user numbers.

1.4 Figure 1 shows the overall flow through the five programs in the FRT model. Also shown are the various data files either read or written during the course of a run through all the computer programs. A brief description of all computer programs and data files utilized by the automated FRT model is provided in Table 2.

ORGANIZATION OF MANUAL

1.5 The remaining portion of this manual describes each of the five computer programs. For each program, a detailed verbal description, flow charts, variable dictionary, routine dictionary (briefly describing the function of each main program and its subroutines), and program listing are provided. In addition, the program listing for the utility program is included.

1.6 Wherever possible, variable names were selected as mnemonics. For example, in CRAWM, the variable LEVLSR denotes the level of complexity within the LSR Generator. In CRAW2, the variable (COSTFH) denotes the (cost per flight hour), etc.

TABLE 1

FRT MODEL PROGRAMS

Name	Length	Storage Units
Source Programs		
CRAWM	1,588 characters	2
CRAWH	5,288 characters	5
CRAW1	14,796 characters	12
CRAW2	5,196 characters	5
CRAW3	11,728 characters	10
XHUNT	2,300 characters	2
Compiled Programs and Data Files		
CRAWM*	2,392 words	2
CRAWH*	3,544 words	6
CRAW1*	6,232 words	14
CRAW2*	3,504 words	5
CRAW3*	5,280 words	11
XHUNT*	848 words	3
SQUAD*1	36,184 characters	29
SQUAD*2	40,104 characters	32

1/ One storage unit contains 1,260 characters.

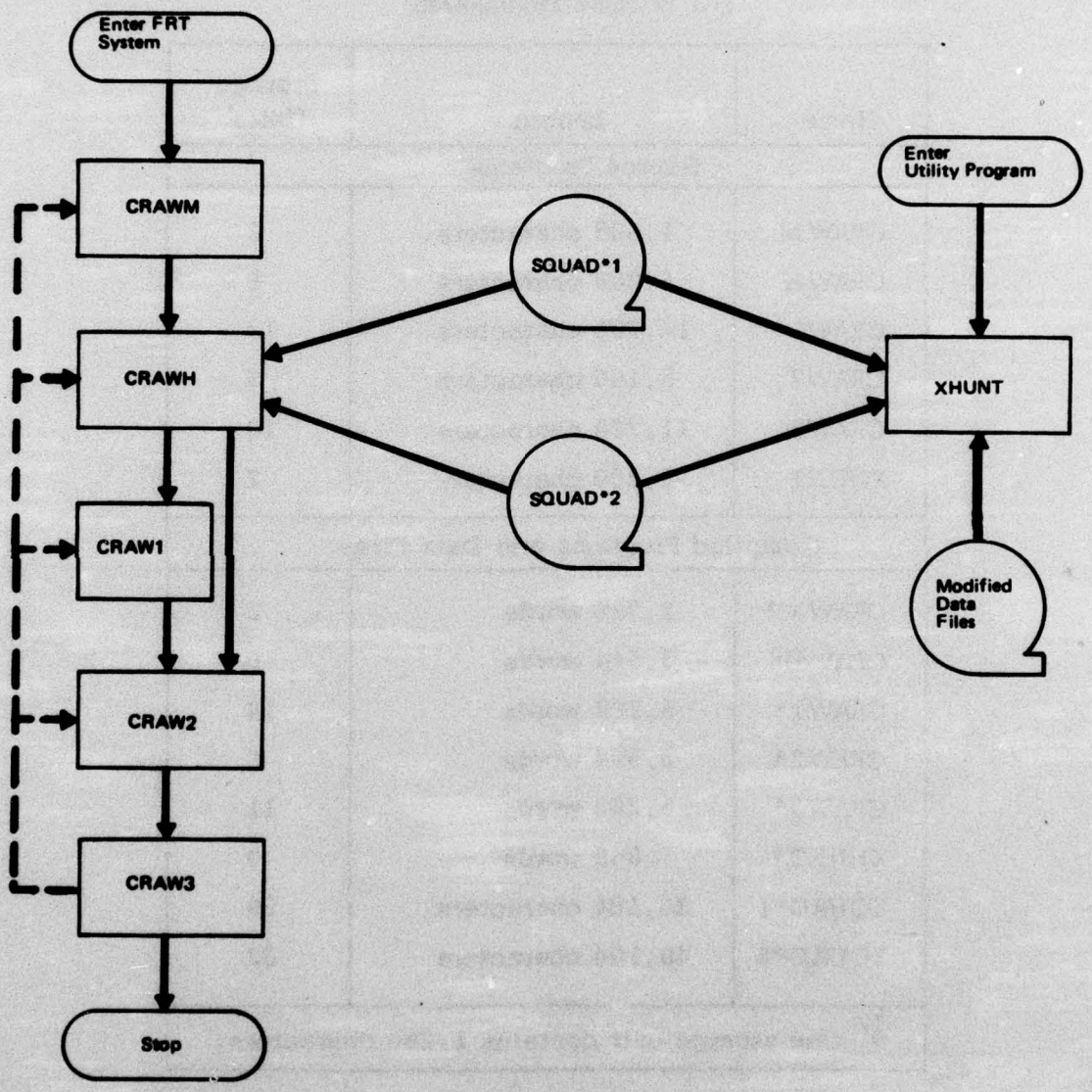


FIGURE 1. OVERALL FLOW THROUGH FRT MODEL

TABLE 2

COMPUTER PROGRAMS

Name	Type	Purpose/Function
CRAWM	Program	Enters level of complexity and annual data for model
CRAWH	Program	Reads in all squadron names then reads in all data for the chosen squadron
CRAW1	Program	Allows user to list and modify planning factors
CRAW2	Program	Accepts throughput data, calculates student load, starts category summary printout
CRAW3	Program	Computes aircraft and instructors required, completes printout of category summary and prints out squadron summary
XHUNT	Program	Utility program to list parts of data files
SQUAD*1	Data file	Contains squadron planning factors
SQUAD*2	Data file	Contains squadron planning factors

II. PROGRAM CRAWM

PROGRAM DESCRIPTION

2.1 The purpose of program CRAWM is to ask questions 1 and 2 and validate the responses.

2.2 Program CRAWM is the first program to be run on the automated FRT model. It may also be entered from CRAW3 when the user chooses to return to question 1.

2.3 The first thing CRAWM does when entered is increment IS(1) by 1. When the user first enters the program from the terminal, all variables are initialized to zero by the Mark II FORTRAN system. Then, if IS(1) = 1 (indicating the first time question 1 is to be asked), the entire question is printed. If IS(1)≥2 (indicating a return from program CRAW3), a new heading, the run number and the first line only of question 1 are printed. The values for level of complexity (LEVLSR), annual training weeks (WPY), and annual fly days (AFD) are entered. The values are checked, and then control is transferred to program CRAWH.

2.4 A flow chart of program CRAWM is shown in Figure 2. Table 3 contains the variable dictionary of program CRAWM. The program dictionary is provided in Table 4, and the program listing is shown in Table 5.

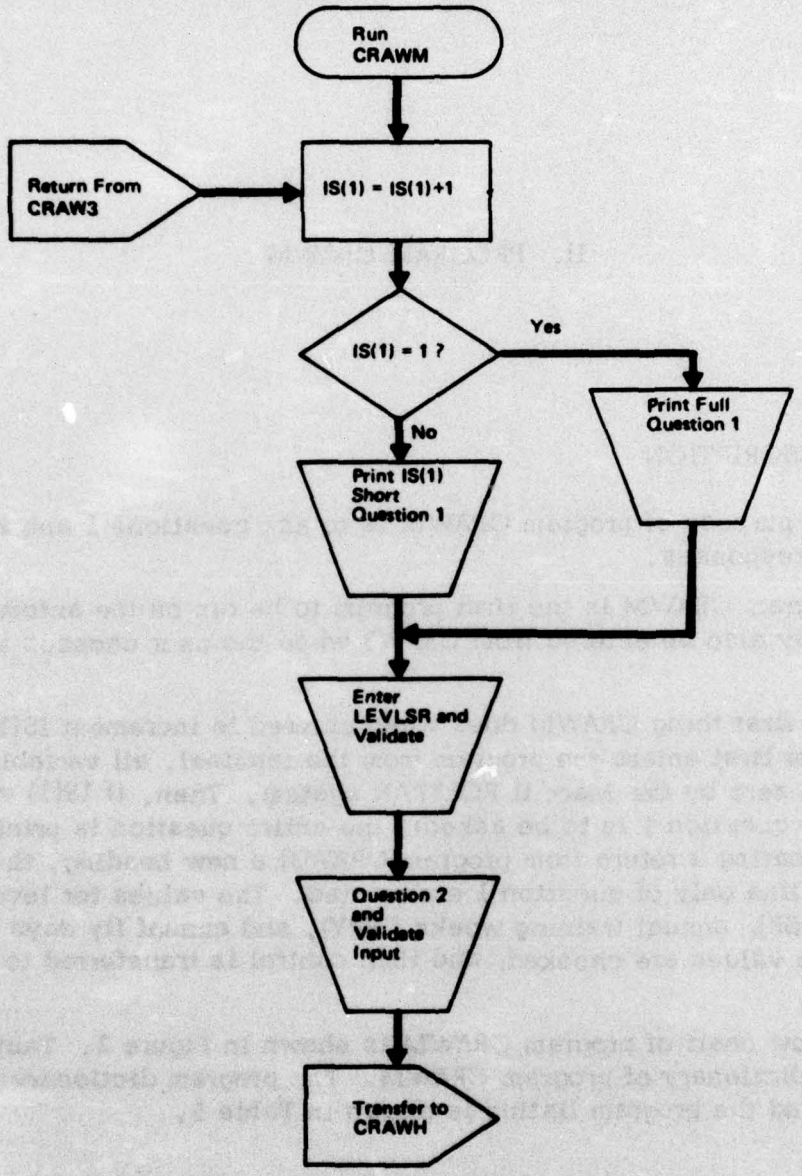


FIGURE 2. PROGRAM CRAWM FLOW CHART

TABLE 3

PROGRAM CRAWM VARIABLE DICTIONARY
(All are located in common)

Variable Name	Dimension	Description
IY	1	Not used
ISW ISW=0 ISW=1	1	Squadron summary print option Print squadron summary Skip squadron summary
LEVLSR	1	Level of complexity
IS IS(1) IS(2)	7	Indicators for other parts of the model The number of times question 1 has been asked Indicates a return from question 8 IS(3) to IS(7) not used
KILL	1	Not used
IBC	1	Not used
NO	1	Character "N"
YES	1	Character "Y"
SW	2	Not used
AFD	1	Annual fly days per year
WPY	1	Training weeks per year
ALLSQD	30, 2	Names of Ith squadrons in data files (2 words, 8 characters permitted)*
NFILE1	1	Number of squadrons stored on data file SQUAD*1
NSQD	1	Total number of squadrons on both data files
SQNAM	2	Chosen squadron name
NCAT	1	Number of categories in squadron
F	2, 10	Squadron personnel data
NAME	25, 3	Name of training category I (3 words, 12 characters)
NPLA	25, 3	Name of aircraft types in category I, J = 1,3*

TABLE 3 (Cont)

Variable Name	Dimension	Description
NFUEL	25, 3	Fuel type for category I, aircraft type J
NAC	25	Number of aircraft types in category I (maximum of 3)
INSTMIX	25, 9	Instruction mix for category I (for J = 1,2,3 first aircraft type; J = 4,5,6 second aircraft type, etc.)
AMO	25, 3	Enlisted maintenance personnel per aircraft type J in category I
WX	25, 3	Percent flyable weather for aircraft type J in category I
PHADUR	25	Duration of category I (weeks)
ATR	25	Attrition rate of category I
ATP	25	Attrition point of category I
GAS	25, 3	Fuel consumption rate for aircraft type J in category I
COSTFH	25, 3	Cost per flight hour for aircraft type J in category I
FITOD	25, 3	Flight instructor tour of duty for aircraft type J in category I (months)
FITR	25, 3	Flight instructor training time for aircraft type J in category I (months)
ACFD	25, 3	Aircraft flight hours per day for aircraft type J, category I
ACHS	25, 3	Aircraft flight hours required per student output in category I on aircraft type J
FINU	25, 3, 3	Flight instructor utilization (hours/day) in category I, aircraft type J, instruction type K
FINHS	25, 3, 3	Flight instructor hours required per student output in category I on aircraft type J, instruction type K
RLSO	25, 3, 3	ACD/LSO/WST ratio in category I for aircraft type J, instruction type K

* I refers to row dimension, and J refers to column dimension.

TABLE 4

CRAWM PROGRAM DICTIONARY

CRAWM	Main program in FRT model. Initializes program variable.

TABLE 5

PROGRAM CRAWM LISTING

```
100C--- PROGRAM: CRAWM   (MAIN PROG)
120       COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
140       COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
160C - - - SQUADRON VARIABLES - - -
180       COMMON SQNAM(2),NCAT,F(2,10)
200C - - -CATAGORY VARIABLES - - -
220       COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9)
240       COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
260      &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
280      &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
300C
320       IS(1)=IS(1)+1
340       KILL=0
360       IBC=0
380     5 IF(IS(1).EQ.1)PRINT 690
400       IF(IS(1).GT.1)PRINT 693,IS(1)
420       PRINT 700
440       IF(IS(1).EQ.1)PRINT 701
460    10 INPUT,LEVLSR
480       IF(LEVLSR)30,30,20
500    20 IF(LEVLSR-3)40,40,30
520    30 PRINT 702
540       GO TO 10
560    40 PRINT 703
580    50 INPUT,WPY,AFD
600       IF(WPY)90,90,60
620    60 IF(WPY-52.)70,70,90
640    70 IF(AFD)90,90,80
660    80 IF(AFD-365.)100,100,90
680    90 PRINT 702
700       GO TO 50
720   100 CHAIN"CRAWH*"
740   700 FORMAT(" Q-1.  ENTER LEVEL OF COMPLEXITY")
760   701 FORMAT("     1 LIMITED SET OF QUESTIONS"/
780      &"     2 DETAILED SET OF QUESTIONS"/
800      &"     3 LIST AND MODIFY PLANNING FACTORS")
820   703 FORMAT(" Q-2.  ENTER TRAINING WEEKS PER YEAR"/
840      &"   AND ANNUAL FLY-DAYS (XX.,XXX.)")
860   702 FORMAT(" INVALID REPLY - RETYPE")
880   690 FORMAT(5X,"FLEET READINESS TRAINING (FRT)"//)
900   693 FORMAT(//10(4H - -)//5X,"  FRT RUN NO.",I3//)
920       END
```

III. PROGRAM CRAWH

PROGRAM DESCRIPTION

3.1 The purpose of program CRAWH is to read into memory the data associated with a given squadron and all training categories. Upon entry to program CRAWH the values of IS(1) and IS(2) are checked.

3.2 If IS(1) = 1 and IS(2) = 0, this indicates the first entry in CRAWH during this run. In this case question 3 is asked if LEVLSR $\neq$ 1 and subroutine READSQ is called. The subroutine puts the squadron names from both data files into the array ALLSQD. After returning from READSQ question 4 is asked.

3.3 If IS(1) $\geq$ 2, or IS(2) $\geq$ 1, this indicates an entry to CRAWH from CRAW3. In this case the array ALLSQD is already set up and the user has had a chance to see the squadron names. The program goes directly to question 4.

3.4 Question 4 asks the user to choose a squadron by typing in the name. Then the array ALLSQD is searched for this name. If it is not found, an error message is printed and the user retypes the name. When the entered squadron name matches a squadron name in ALLSQD, the program determines on which data file the squadron is located by comparing its location in ALLSQD with NFILE1, the number of squadrons in SQUAD*1. The required number of squadrons to be skipped in the data file is also calculated. The data file is then opened and set to the squadron selected by the user. (Note that the data file is sequential and all records must be read in order to skip them.) Next subroutine DATAIN is called which reads the data for the selected squadron and training categories. Upon returning to CRAWH the data file is closed.

3.5 Control is transferred to program CRAW1 if LEVLSR (level of complexity) is 3. Otherwise, control is transferred to program CRAW2.

SUBROUTINE READSQ

3.6 The purpose of subroutine READSQ is to read the squadron names from both data files (SQUAD*1 and SQUAD*2) and put the names into the array ALLSQD. It also counts the number of squadrons in SQUAD*1, which is saved in NFILE1. It is called once during each FRT model run. The subroutine determines the end of the data file by testing the squadron name for the four characters "END$." Also only 30 squadron names are read. Then if K, the argument of subroutine READSQ, is zero, control is returned to CRAWH. If K is not zero, the squadron names are printed.

SUBROUTINE NOYES

3.7 The purpose of subroutine NOYES is to read and validate a "no" (N), or "yes" (Y), response from the terminal. If the response is valid, the appropriate nonstandard return is taken. The nonstandard return transfers control to the proper statement in the calling program. Return 1 is taken for a no response, Return 2 is taken for a yes response. If the response is invalid, the user must retype it.

SUBROUTINE DATAIN

3.8 The purpose of subroutine DATAIN is to read the data for the selected squadron. First, it reads the squadron name and number of categories. Then the squadron support personnel are read into the F array. Finally the data for all categories are read.

VARIABLE DICTIONARY

3.9 Not all variables are listed in Table 7, since some are just used for indexing or their use is obvious. No new common variables have been added in this program.

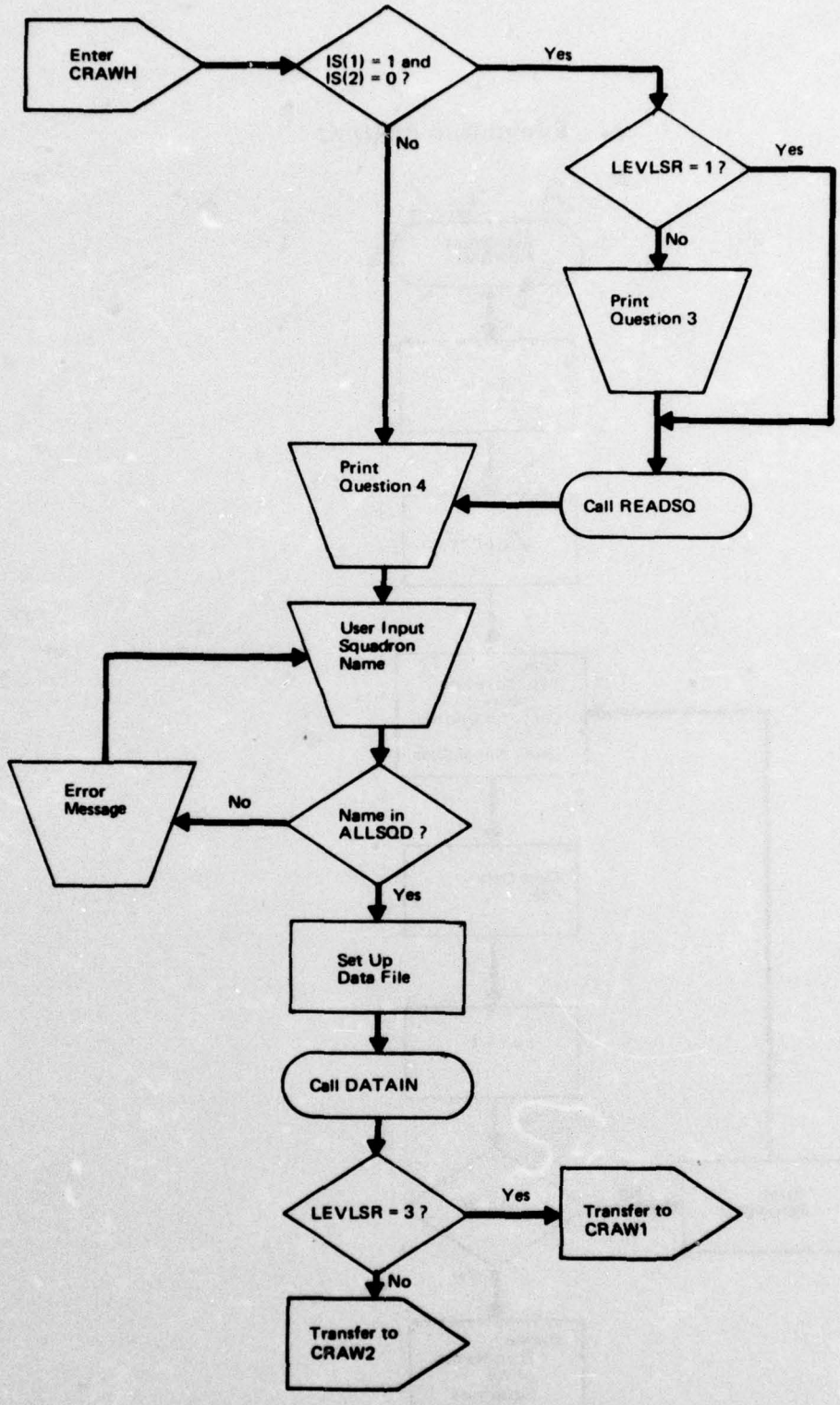


FIGURE 3. PROGRAM CRAWH FLOW CHART

a. Subroutine READSQ

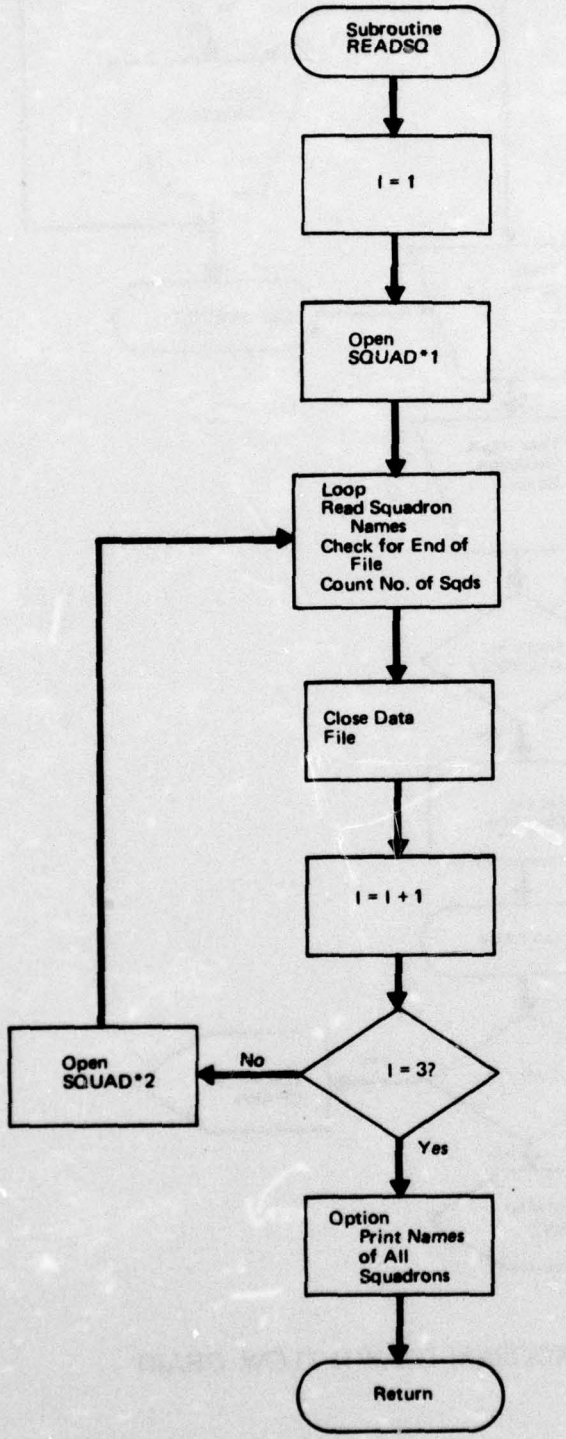


FIGURE 3 (Cont)

b. Subroutine NOYES

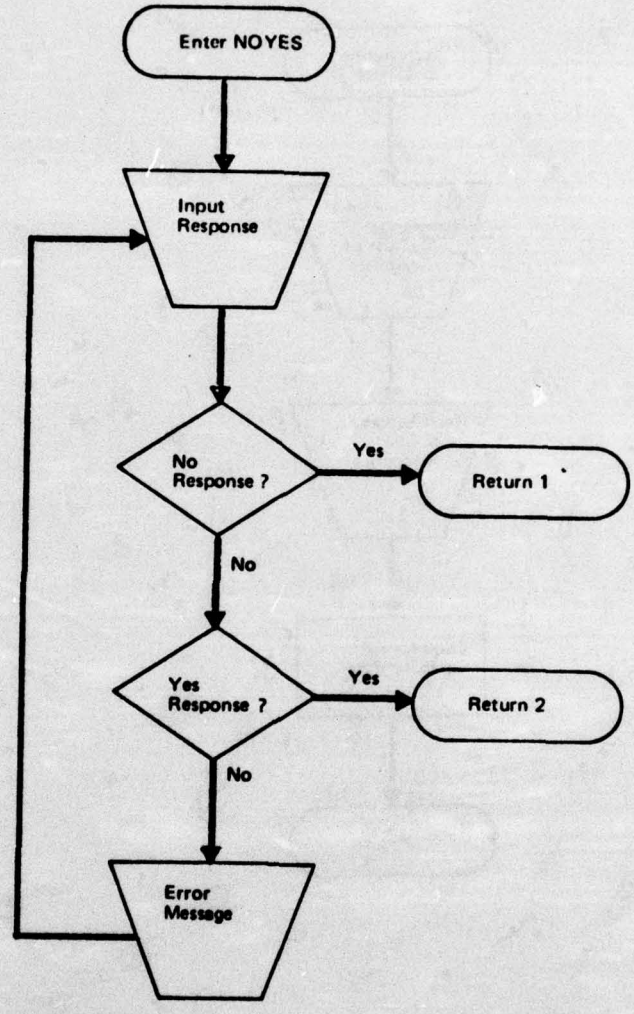


FIGURE 3 (Cont)

c. Subroutine DATAIN

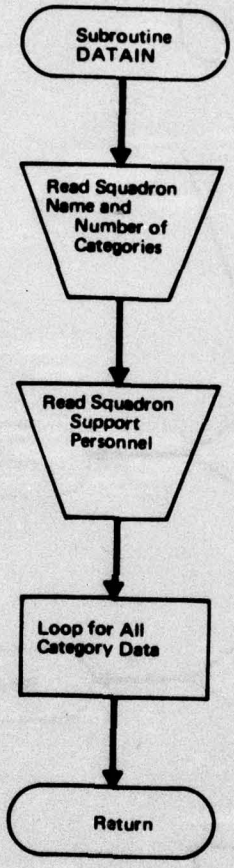


FIGURE 3 (Cont)

TABLE 6

CRAWM PROGRAM AND SUBROUTINE DICTIONARY

CRAWH	Main routine, prints questions 3 and 4. Sets up data file for DATAIN routine.
READSQ	Reads data files for names and number of squadrons.
NOYES	Reads a no "N", or yes "Y" from terminal, provides branching.
DATAIN	Reads squadron and category data.

TABLE 7

PROGRAM CRAWH VARIABLE DICTIONARY

Location	Variable Name	Dimension	Description
CRAWH	EOF	1	Characters "END$\not$"
READSQ	EOF	1	Characters "END$\not$"
CRAWH	T1	1	File name
READSQ	T1	1	File name
DATAIN	T1	1	File name
READSQ	K	1	K = 0, do not print squadron names K = 1, print squadron names
CRAWH	K	1	Squadron number location indicator
CRAWH	KK	1	Data file line number
READSQ	KK	1	Data file line number
READSQ	DUM	14	First 14 locations in common
DATAIN	DUM	14	First 14 locations in common
NOYES	DUM	12	First 12 locations in common

TABLE 8

PROGRAM CRAWH LISTING

```
106C---PROGRAM: CRAWH (READS DATA FILES)
126       COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
146       COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
166C - - - SQUADRON VARIABLES - - -
186       COMMON SQNAM(2),NCAT,F(2,10)
206C - - -CATEGORY VARIABLES - - -
226       COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9)
246       COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
266      &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
286      &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
306C - - -
326       ALPHA NO,YES,ALLSQD,SQNAM
346       ALPHA EOF
366       FILENAME T1
386       NO="N"
406       YES="Y"
426       EOF="END "
446C
466       IF(IS(2).EQ.1)GO TO 140
486       IF(IS(1).GT.1)GO TO 140
506       IF(LEVLSR.EQ.1)GO TO 110
526       PRINT 710
546       CALL NOYES($110,$105)
566   105 CALL READSQ(1)
586       GO TO 140
606   110 CALL READSQ(0)
626   140 PRINT 715
646   145 INPUT 905,SQNAM
666C - - CHECK FOR VALID NAME ; WHICH FILE ? ; FIND IT.
686   150 DO 160 I=1,NSQD
706       IF(ALLSQD(I,1).NE.SQNAM(1))GO TO 160
726       IF(ALLSQD(I,2).NE.SQNAM(2))GO TO 160
746       K=I
766       GO TO 170
786   160 CONTINUE
806       PRINT 720
826       GO TO 145
```

TABLE 8 (Cont)

```
846   170 T1="SQUAD*1"
866       IF(K.GT.NFILE1) T1="SQUAD*2"
886       IF(K.GT.NFILE1)K=K-NFILE1
906C  - - FIND THE SQUAD ON FILE T1
926       OPENFILE T1 ; REWIND T1
946       K=K-1
966       IF(K.EQ.0)GO TO 500
986       DO 180 I=1,K
1006       READ(T1,908)KK
1026       M=20*KK+2
1046C - - - THE 2 IS FOR TWO LINES OF SQUAD DATA
1066       DO 180 J=1,M
1086  180 READ(T1,910)KK
1106       GO TO 500
1126C - - - READ SQUAD AND ALL CAT DATA FROM T1. - -
1146  500 CALL DATAIN(T1)
1166       CLOSEFILE T1
1186       IF(LEVLSR.LE.2)CHAIN"CRAW2*"
1206       CHAIN"CRAW1*"
1226C
1246  705 FORMAT(" INVALID REPLY - RETYPE")
1266  710 FORMAT(" Q-3.  PRINT NAMES OF ALL SQUADRONS(Y,N)")
1286  715 FORMAT(" Q-4.  ENTER SQUADRON NAME (AAAAAAAA)")
1306  720 FORMAT("  SQUADRON DOES NOT EXIST - RETYPE")
1326C
1346  905 FORMAT(2A4)
1366  908 FORMAT(13X,I4)
1386  910 FORMAT(I4)
1406  915 FORMAT(5X,3A4,2I3)
1426  920 FORMAT(3A4)
1446  930 FORMAT(5X,5A4,1X,5A4)
1466       END
```

TABLE 8 (Cont)

a. Subroutine READSQ

```
1486        SUBROUTINE READSQ(K)
1506        COMMON DUM(14)
1526        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
1546        DIMENSION SQ(2)
1566        ALPHA EOF,ALLSQD,SQ
1586        FILENAME T1
1606        NSQD=1
1626        EOF="END "
1646        I=1
1666        T1="SQUAD*1"
1686     10 OPENFILE T1
1706        REWIND T1
1726     15 READ(T1,700)SQ,N
1746        IF(SQ(1).EQ.EOF)GO TO 50
1766        IF(I.EQ.31)GO TO 50
1786        ALLSQD(I,1)=SQ(1)
1806        ALLSQD(I,2)=SQ(2)
1826        N=20*N+2
1846        DO 30 J=1,N
1866     30 READ(T1,705)KK
1886        I=I+1
1906        GO TO 15
1926C    - - -
1946     50 NSQD=NSQD+1
1966        CLOSEFILE T1
1986        IF(NSQD.EQ.3)GO TO 100
2006        NFILE1=I-1
2026        T1="SQUAD*2"
2046        GO TO 10
2066C
2086    100 NSQD=I-1
2106        IF(K.EQ.0)RETURN
2126        PRINT 710,( (ALLSQD(I,J),J=1,2),I=1,NSQD)
2146        PRINT,"  "
2166    700 FORMAT(5X,2A4,I4)
2186    705 FORMAT(I4)
2206    710 FORMAT(5X,2A4,5X,2A4,5X,2A4)
2226        RETURN$END
```

TABLE 8 (Cont)

b. Subroutine NOYES

```
2246        SUBROUTINE NOYES(*,*)
2266        COMMON DUM(12),NO,YES
2286        ALPHA NO,YES,N
2306     10 INPUT 700,N
2326        IF(N.EQ.NO)RETURN1
2346        IF(N.EQ.YES)RETURN2
2366        PRINT," INVALID REPLY - RETYPE"
2386        GO TO 10
2406    700 FORMAT(A1)
2426        END
2446C
```

TABLE 8 (Cont)

c. Subroutine DATAIN

```
2456        SUBROUTINE DATAIN(T1)
2486        COMMON DUM(14)
2506        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
2526        COMMON SQNAM(2),NCAT,F(2,10)
2546        COMMON NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9)
2566        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
2586       & GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
2606       & ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
2626        FILENAME T1
2646C
2666        READ(T1,3)SQNAM,NCAT
2686      3 FORMAT(5X,2A4,I4)
2706      5 FORMAT(V)
2726     10 FORMAT(I4,1X,9A4)
2746        READ(T1,5)IL,(F(1,J),J=1,5)
2766        READ(T1,5)IL,(F(2,J),J=1,5)
2786        DO 500 I=1,NCAT
2806        READ(T1,10)IL,(NAME(I,J),J=1,3),(NPLA(I,J),J=1,3),
2826       &(NFUEL(I,J),J=1,3)
2846        READ(T1,5)IL,NAC(I),(INSTMIX(I,J),J=1,9)
2866        READ(T1,5)IL,(AMO(I,J),J=1,3),(WX(I,J),J=1,3)
2886        READ(T1,5)IL,PHADUR(I),ATR(I),ATP(I)
2906        READ(T1,5)IL,(GAS(I,J),J=1,3),(COSTFH(I,J),J=1,3)
2926        READ(T1,5)IL,(FITOD(I,J),J=1,3),(FITR(I,J),J=1,3)
2946        READ(T1,5)IL,(ACFD(I,J),J=1,3),(ACHS(I,J),J=1,3)
2966C - - FLIGHT INSTRUCTOR UTILIZATION
2986        DO 50 K=1,3
3006     50 READ(T1,5)IL,(FINU(I,K,J),J=1,3)
3026C - - -FLIGHT INSTR  HRS/STUD
3046        DO 60 K=1,3
3066     60 READ(T1,5)IL,(FINHS(I,K,J),J=1,3)
3086C - -  LSO/WST RATIO
3106        DO 70 K=1,3
3126     70 READ(T1,5)IL,(RLSO(I,K,J),J=1,3)
3146C - - READ 4 SPARE LINES (IL IS NOT USED)
3166        DO 80 K=1,4
3186     80 READ(T1,5)IL
3206    500 CONTINUE
3226C - - NO CHECKS ON DATA HAVE BEEN PERFORMED
3246        RETURN;END
```

IV. PROGRAM CRAW1

PROGRAM DESCRIPTION

4.1 The purpose of program CRAW1 is to allow the user to list and modify the data that were read by program CRAWH or previously entered by this program. These changes do not permanently change the data files and are not checked for validity, e.g., percentages are not checked for being positive and less than 1.0.

4.2 First subroutine SQUAD is called so the user can list and modify the squadron support personnel. Then the option to list the planning factors for categories is given. All the planning factors for a category are listed by a loop which repeatedly calls subroutine CATOUT. One of the arguments is the planning factor item number to be listed. The item number is incremented by 1 in the loop; thus all planning factors are printed. Next single planning factors can be listed. This is done by calling subroutine CATOUT with the desired category number and item number.

4.3 When no further planning factors are to be listed, then question 25A allows the user to modify planning factors. The change is accepted by calling subroutine CATMOD which prints out the planning factor name and the number of data entries, depending on the number of aircraft.

4.4 After all changes to planning factors in existing categories are made, the user may add a new category. If a new category is added, subroutine CATMOD is called 25 times—once for each different planning factor item number. Next, if the user wants the planning factors listed for the new category, the program goes through another loop calling subroutine CATOUT for all planning factors. Again, individual planning factors may be changed for the new category. This is done by calling CATMOD. If the user wants to add another category, the above procedure is repeated. When no further categories are to be added, control is transferred to program CRAW2.

SUBROUTINE SQUAD

4.5 Subroutine SQUAD asks questions 21 and 22. It allows the user to print or change the squadron support personnel data. These data are stored in the F array. Some calculation is required by the program to convert the input planning factor item number (1 through 10) into the proper subscripts for the F array.

SUBROUTINE NOYES

4.6 The purpose of subroutine NOYES is to read and validate a no (N), or yes (Y), response from the terminal. If the response is valid, the appropriate nonstandard return is taken. The nonstandard return transfers control to the proper statement in the calling program. Return 1 is taken for a no response, Return 2 is taken for a yes response. If the response is invalid, the user must retype it.

SUBROUTINE CHECK

4.7 This subroutine checks to see if an integer is in a given range. If it is, the standard return is taken and the program continues. If the value is out of range, an error message is printed, and the program returns to the previous input statement by the use of a nonstandard return.

SUBROUTINE CATOUT

4.8 The purpose of the subroutine is to print out the planning factor name and value, or values if a category contains more than one aircraft type. The arguments in subroutine CATOUT are the category number and the planning factor item (reference) number. This subroutine is called in the loop to list all planning factors for a category. It is also called when single items are to be listed.

SUBROUTINE CATMOD

4.9 The purpose of subroutine CATMOD is to print out the name of the planning factor item and the number of values to be entered and to accept the new values. The arguments in subroutine CATMOD are the category number and planning factor item number. Subroutine CATMOD is called inside a loop when a new category is added. It is also called when single items are to be changed.

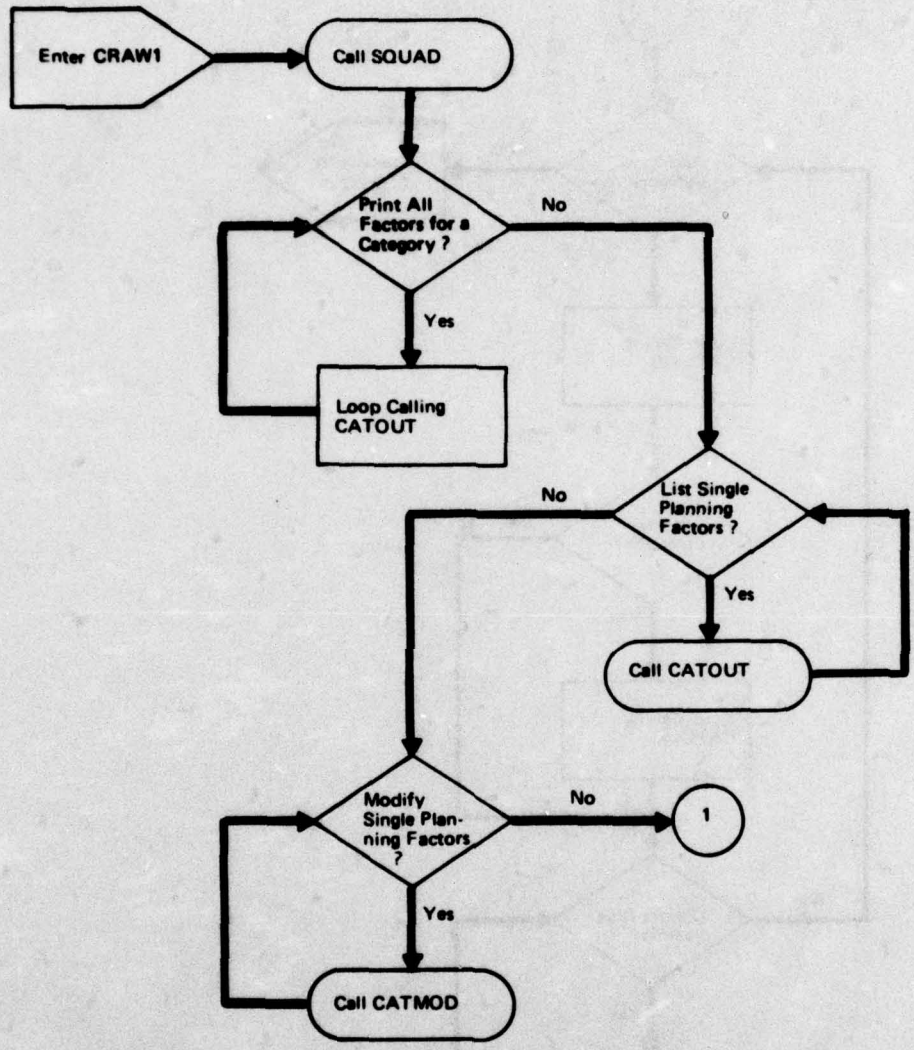


FIGURE 4. PROGRAM CRAW1 FLOW CHART

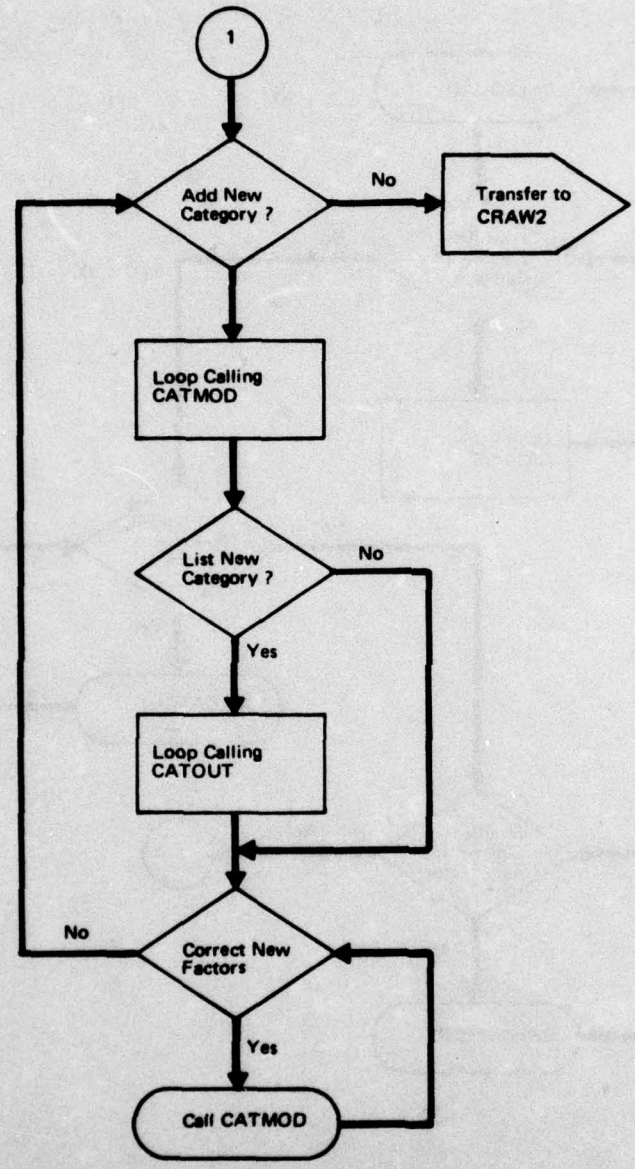


FIGURE 4 (Cont)

a. Subroutine SQUAD

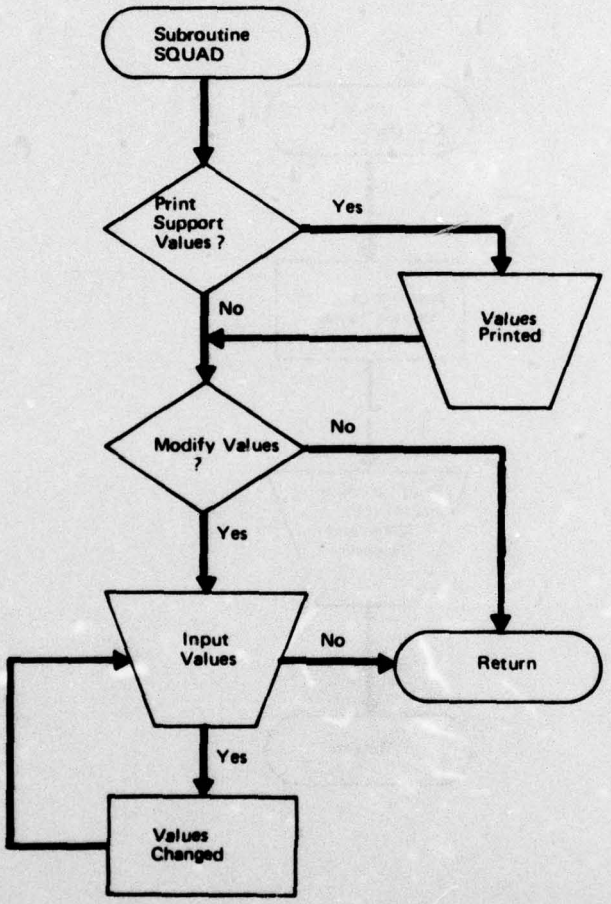


FIGURE 4 (Cont)

b. Subroutine CATOUT

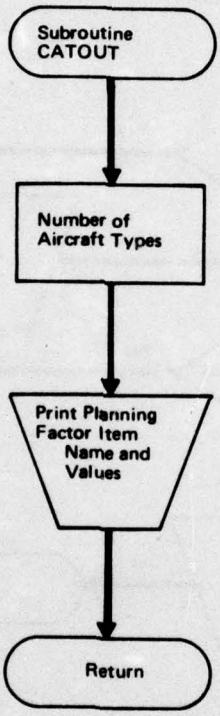


FIGURE 4 (Cont)

c. Subroutine CATMOD

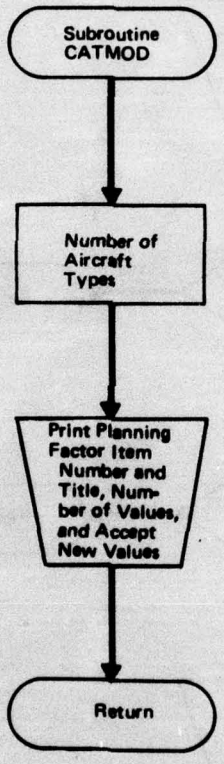


FIGURE 4 (Cont)

d. Subroutine NOYES

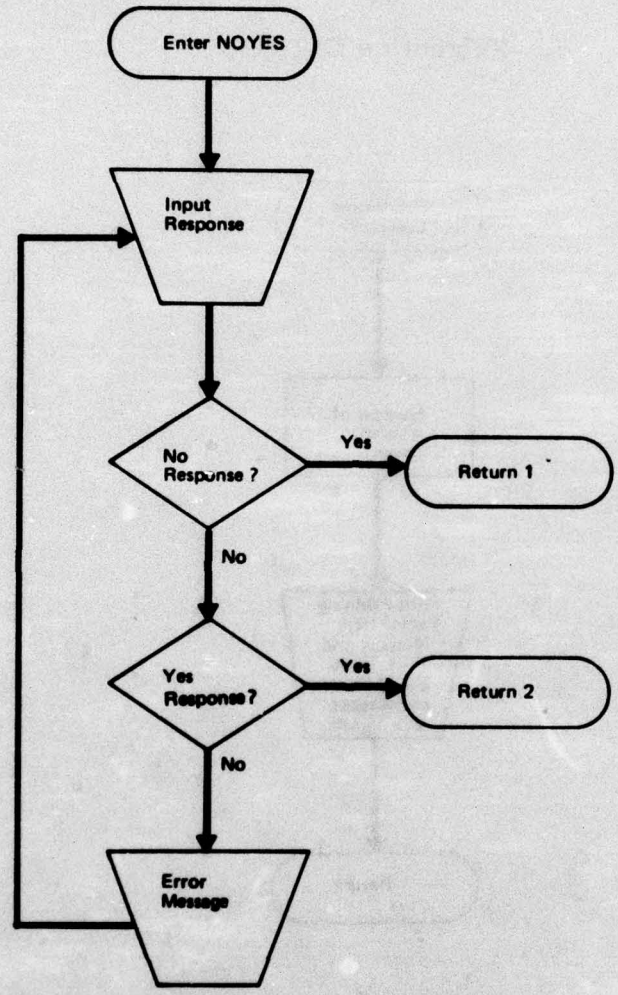


FIGURE 4 (Cont)

e. Subroutine CHECK

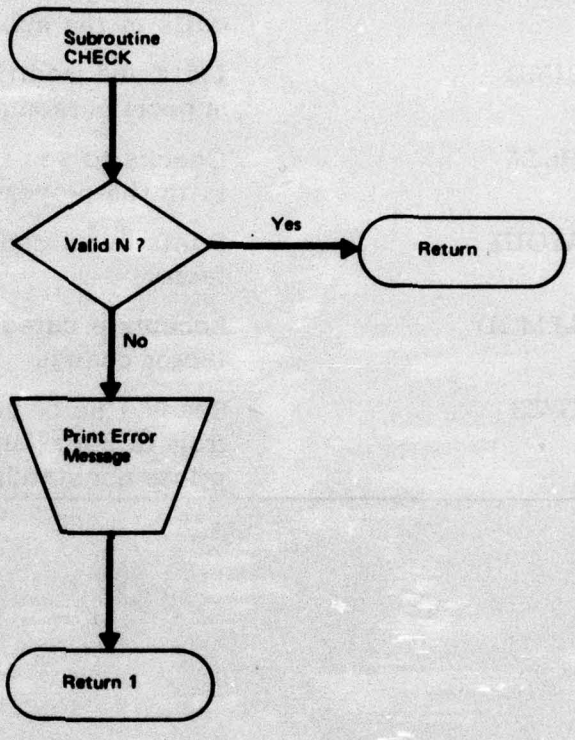


FIGURE 4 (Cont)

TABLE 9

CRAW1 PROGRAM AND SUBROUTINE DICTIONARY

CRAW 1	Main program to control the calls to the subroutines
SQUAD	Lists and modifies squadron support personnel
CHECK	Checks to see if the response is in the proper range
CATOUT	Prints out a category planning factor
CATMOD	Accepts a category planning factor change
NOYES	Reads a no or yes (N or Y) from terminal and takes appropriate nonstandard return

TABLE 10

PROGRAM CRAW1 VARIABLE DICTIONARY

Location	Name	Description
CRAW1	NP	Current number of planning factors
CRAW1	ICAT	Category number—user input
CRAW1	I	Category planning factor reference number
SQUAD*	N	Squadron personnel planning factor reference number—user input
SQUAD	V	New value for planning factor—user input
NOYES	DUM	Equivalent to the first 12 locations in common
CHECK	DUM	Equivalent to the first 12 locations in common
CATOUT	JC	Category number
CATOUT	I	Category planning factor reference number
CATOUT	N	Number of type of aircraft
CATMOD	T	Characters "ENTER:Ø"
CATMOD	IN	3 words, characters "IP," "INFO," "IC/N"
CATMOD	IC	Category number
CATMOD	I	Category planning factor reference number

TABLE 11

PROGRAM CRAW1 LISTING

```
101C--- PROGRAM: CRAW1 (PRINT-MODIFY DATA)
121       COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
141       COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
161C - - - SQUADRON VARIABLES - - -
181       COMMON SQNAM(2),NCAT,F(2,10)
201C - - -CATEGORY VARIABLES - - -
221       COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9)
241       COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
261      &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
281      &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
301C - - -
321       COMMON SI(25),SO(25),SL(25),ATRI(25)
341C
361       CALL SQUAD
381C
401       NP=25
421       PRINT 600
441    25 INPUT,ICAT
461       IF(ICAT.EQ.0)GO TO 40
481       CALL CHECK(ICAT,1,NCAT,$25)
501    30 DO 35 I=1,NP
521       IF(I.EQ.8)PRINT," "
541    35 CALL CATOUT(ICAT,I)
561       PRINT 610
581       GO TO 25
601C
621    40 PRINT 615
641    45 INPUT,ICAT,I
661       IF(ICAT.EQ.0)GO TO 60
681       CALL CHECK(ICAT,1,NCAT,$45)
701       CALL CHECK(I,1,NP,$45)
721       CALL CATOUT(ICAT,I)
741       PRINT 620
761       GO TO 45
781C - - - - -
```

TABLE 11 (Cont)

```
801    60 PRINT 630
821    65 INPUT,ICAT,I
841       IF(ICAT.EQ.0)GO TO 80
861       CALL CHECK(ICAT,1,NCAT,$65)
881       CALL CHECK(I,1,NP,$65)
901       CALL CATMOD(ICAT,I)
921       PRINT 640
941       GO TO 65
961C
981    80 PRINT 650
1001    85 CALL NOYES($150,$90)
1021    90 IF(NCAT.EQ.25)GO TO 130
1041       NCAT=NCAT+1
1061       DO 100 I=1,NP
1081   100 CALL CATMOD(NCAT,I)
1101       PRINT 700
1121       CALL NOYES($120,$105)
1141   105 DO 108 I=1,NP
1161       IF(I.EQ.8)PRINT," "
1181   108 CALL CATOUT(NCAT,I)
1201C
1221   120 PRINT 710
1241   121 INPUT,I
1261       IF(I.EQ.0)GO TO 125
1281       CALL CHECK(I,1,NP,$121)
1301       CALL CATMOD(NCAT,I)
1321       PRINT 720
1341       GO TO 121
1361C
1381   125 PRINT 660
1401       GO TO 85
1421C
1441   130 PRINT 670
1461   150 CHAIN "CRAW2*"
1481C
```

TABLE 11 (Cont)

```
1501   600 FORMAT(/" Q-23A. FOR DETAILED LIST OF ALL CATEGORY",
1521      &" PLANNING FACTORS"/"    ENTER THE CAT. NO. (XX)"/
1541      &"    (ENTER 0 FOR NO FURTHER DETAIL) ")
1561   610 FORMAT(/" Q-23B.  TO LIST PLANNING FACTORS FOR ANOTHER",
1581      &" CATEGORY"/"    ENTER CAT. NO.(OR 0) ")
1601   615 FORMAT(/" Q-24A.  TO LIST A SPECIFIC PLANNING FACTOR"/
1621      &"    ENTER THE CAT. NO. AND THE PLANNING FACTOR",
1641      &" NO.(XX,XX)"/"    ENTER 0,0 FOR NO FURTHER DETAIL  ")
1661   620 FORMAT(/" Q-24B.  NEXT PLANNING FACTOR(XX,XX)")
1681   630 FORMAT(/" Q-25A.  TO MODIFY A PLANNING FACTOR"/
1701      &"   ENTER THE CAT. NO. AND THE PLANNING FACTOR NO. (XX,XX)"
1721      &/"   (ENTER 0,0 FOR NO MODIFICATIONS) " )
1741   640 FORMAT(/" Q-25B. NEXT MODIFICATION(XX,XX)")
1761   650 FORMAT(/" Q-26A.  DO YOU WANT TO ADD A CATEGORY(Y,N)")
1781   660 FORMAT(/" Q-26B.  ADD ANOTHER CATEGORY(Y,N)")
1801   670 FORMAT(//"  NO SPACE FOR AN ADDITIONAL CATEGORY"/
1821      &"  ONLY 25 CATEGORIES IN A SQUADRON"/
1841      &"  *  PROGRAM CONTINUES *"///)
1861C
1881   700 FORMAT(/" Q-27. DO YOU WANT A LISTING OF ALL PLANNING",
1901      &" FACTORS"/"    FOR THE NEW CATEGORY(Y,N)")
1921   710 FORMAT(/" Q-28A. TO CHANGE OR CORRECT A PLANNING FACTOR",
1941      &/"    FOR THE NEW CAT. ENTER THE PLANNING FACTOR NUMBER"/
1961      &"    ENTER 0 FOR NO FURTHER CHANGES ")
1981   720 FORMAT(/" Q-28B. NEXT PLANNING FACTOR (XX)")
2001       END
```

TABLE 11 (Cont)

a. Subroutine SQUAD

```
2021        SUBROUTINE SQUAD
2041        COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
2061        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
2081C - - - SQUADRON VARIABLES - - -
2101        COMMON SQNAM(2),NCAT,F(2,10)
2121C - - -CATEGORY VARIABLES - - -
2141        COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
2161        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
2181       &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
2201       &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
2221C - - -
2241        COMMON SI(25),SO(25),SL(25),ATRI(25)
2261C
2281        PRINT 700
2301        CALL NOYES($100,$50)
2321     50 PRINT 710,(F(1,I),I=1,5)
2341        PRINT 720,(F(2,I),I=1,5)
2361    100 PRINT 730
2381    120 INPUT,N,V
2401        IF(N.EQ.0)GO TO 200
2421        IF( (N.GE.1).AND.(N.LE.10) )GO TO 130
2441        GO TO 140
2461    130 IF(V.GE.0.)GO TO 150
2481    140 PRINT 790
2501        GO TO 120
2521C
2541    150 I1=(N-1)/5+1
2561        I2=N-5*(I1-1)
2581        F(I1,I2)=V
2601        PRINT 740
2621        GO TO 120
2641C
2661    200 RETURN
```

TABLE 11 (Cont)

a. Subroutine SQUAD (Cont)

```
2681  700 FORMAT(/" Q-21.  DO YOU WANT A LIST OF SQUADRON"/
2701     &"    PERSONNEL FACTORS(Y,N)")
2721  710 FORMAT(/"   ADMIN. OFFICERS"//
2741     &" 1 IP    ",F6.2/" 2 INFO  ",F6.2/
2761     &" 3 GROUND",F6.2/" 4 MAINT.GD",F6.2/
2781     &" 5 OTHER ",F6.2)
2801  720 FORMAT(/"   ENLISTED SUPPORT"/
2821     &" 6 TRAIN.",F6.2/" 7 DET.  ",F6.2/
2841     &" 8 SITE  ",F6.2/" 9 ADM.  ".F6.2/
2861     &" 10 CREW ",F6.2)
2881  730 FORMAT(/" Q-22.  TO CHANGE A PERSONNEL FACTOR ENTER"/
2901     &"    ELEMENT NO. AND NEW VALUE(XX,XX.)"
2921     &/"    (ENTER 0,0 FOR NO FURTHER CHANGES)")
2941  740 FORMAT("+NEXT")
2961  790 FORMAT(/" INVALID REPLY - RETYPE")
2981      END
```

b. Subroutine NOYES

```
3001      SUBROUTINE NOYES(*,*)
3021      COMMON DUM(12),NO,YES
3041      ALPHA NO,YES,N
3061   10 INPUT 700,N
3081      IF(N.EQ.NO)RETURN1
3101      IF(N.EQ.YES)RETURN2
3121      PRINT," INVALID REPLY - RETYPE"
3141      GO TO 10
3161  700 FORMAT(A1)
3181      END
```

c. Subroutine CHECK

```
3201      SUBROUTINE CHECK(N,NLO,NUP,*)
3221      COMMON DUM(12)
3241      IF( (N.GE.NLO).AND.(N.LE.NUP) )RETURN
3261      PRINT,"INVALID REPLY - RETYPE"
3281      RETURN1
3301      END
```

TABLE 11 (Cont)

d. Subroutine CATOUT

```
3321        SUBROUTINE CATOUT(IC,I)
3341        COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
3361        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
3381C - - - SQUADRON VARIABLES - - -
3401        COMMON SQNAM(2),NCAT,F(2,10)
3421C - - -CATEGORY VARIABLES - - -
3441        COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
3461        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
3481       &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
3501       &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
3521C - - -
3541        COMMON SI(25),SO(25),SL(25),ATRI(25)
3561        N=NAC(IC)
3581        GO TO(10,20,30,40,50,60,70,80,90,100,110,120,130,140,
3601       &150,160,170,180,190,200,210,220,230,240,250),I
3621    10 PRINT 510,I,(NAME(IC,J),J=1,3)
3641   510 FORMAT(I3," CATEGORY NAME - - - - - - -",3A4)
3661        RETURN
3681    20 PRINT 520,I,N
3701   520 FORMAT(I3," NUM. OF TYPES OF AIRCRAFT  ",I3)
3721        RETURN
3741    30 PRINT 530,I,PHADUR(IC)
3761   530 FORMAT(I3," WEEKS TO COMPLETE TRAINING ",F5.1)
3781        RETURN
3801    40 PRINT 540,I,ATR(IC)
3821   540 FORMAT(I3," ATTRITION RATE(100%=1.)    ",F7.3)
3841        RETURN
3861    50 PRINT 550,I,ATP(IC)
3881   550 FORMAT(I3," ATTRITION POINT",12X,F7.3)
3901        RETURN
3921    60 PRINT 560,I
3941        PRINT 565,(FITOD(IC,J),J=1,3)
3961   560 FORMAT(I3," TOUR OF DUTY FOR - - - - -  IP",
3981      &"    INFO   IC/N")
4001   565 FORMAT(6X,"(MONTHS)",14X,3F7.0)
4021        RETURN
4041    70 PRINT 570,I
4061        PRINT 565,(FITR(IC,J),J=1,3)
4081   570 FORMAT(I3," TRAINING PERIOD FOR- - - -  IP",
4101      &"    INFO   IC/N")
4121        RETURN
```

TABLE 11 (Cont)

d. Subroutine CATOUT (Cont)

```
4141    80 PRINT 580,I,(NPLA(IC,J),J=1,N)
4161   580 FORMAT(I3," AIRCRAFT TYPE",7X,3(5X,A4) )
4181       RETURN
4201    90 PRINT 590,I,(AMO(IC,J),J=1,N)
4221   590 FORMAT(I3," MO FACTOR",12X,3F9.2)
4241       RETURN
4261   100 PRINT 600,I,(WX(IC,J),J=1,N)
4281   600 FORMAT(I3," WEATHER(100%=1.)",6X,3F9.3)
4301       RETURN
4321   110 PRINT 610,I,(NFUEL(IC,J),J=1,N)
4341   610 FORMAT(I3," FUEL TYPE",11X,3(5X,A4) )
4361       RETURN
4381   120 PRINT 620,I,(GAS(IC,J),J=1,N)
4401   620 FORMAT(I3," FUEL CONSUMPTION",5X,3F9.2)
4421       RETURN
4441   130 PRINT 630,I,(COSTFH(IC,J),J=1,N)
4461   630 FORMAT(I3," $ PER FLIGHT HOUR",4X,3F9.2)
4481       RETURN
4501   140 PRINT 640,I,(ACFD(IC,J),J=1,N)
4521   640 FORMAT(I3," A/C FLT. HRS/DAY     ",3F9.2)
4541       RETURN
4561   150 PRINT 650,I,(ACHS(IC,J),J=1,N)
4581   650 FORMAT(I3," A/C HOURS/STUDENT",4X,3F9.2)
4601       RETURN
4621   160 M=3*N
4641       PRINT 660,I,(INSTMIX(IC,J),J=1,M)
4661   660 FORMAT(I3," INSTRUCTION TYPES"/
4681      &3X,"(1=IP,2=INFO,3=IC/N)  ",3(I5,2(",",I1)) )
4701       RETURN
4721   170 PRINT 670,I,(FINU(IC,1,J),J=1,N)
4741   670 FORMAT(I3," IP   UTILIZ.(FLY.DAY)",3F9.2)
4761       RETURN
```

TABLE 11 (Cont)

d. Subroutine CATOUT (Cont)

```
4781   180 PRINT 680,I,(FINU(IC,2,J),J=1,N)
4801   680 FORMAT(I3," INFO UTILIZ.(FLY.DAY)",3F9.2)
4821       RETURN
4841   190 PRINT 690,I,(FINU(IC,3,J),J=1,N)
4861   690 FORMAT(I3," IC/N UTILIZ.(FLY.DAY)",3F9.2)
4881       RETURN
4901   200 PRINT 700,I,(FINHS(IC,1,J),J=1,N)
4921   700 FORMAT(I3," IP    INSTR HRS/STUD. ",3F9.2)
4941       RETURN
4961   210 PRINT 710,I,(FINHS(IC,2,J),J=1,N)
4981   710 FORMAT(I3," INFO INSTR HRS/STUD. ",3F9.2)
5001       RETURN
5021   220 PRINT 720,I,(FINHS(IC,3,J),J=1,N)
5041   720 FORMAT(I3," IC/N INSTR HRS/STUD. ",3F9.2)
5061       RETURN
5081   230 PRINT 730,I,(RLSO(IC,1,J),J=1,N)
5101   730 FORMAT(I3," IP    ACD/LSO/WST RATIO ",F7.2,2F9.2)
5121       RETURN
5141   240 PRINT 740,I,(RLSO(IC,2,J),J=1,N)
5161   740 FORMAT(I3," INFO ACD/LSO/WST RATIO ",F7.2,2F9.2)
5181       RETURN
5201   250 PRINT 750,I,(RLSO(IC,3,J),J=1,N)
5221   750 FORMAT(I3," IC/N ACD/LSO/WST RATIO ",F7.2,2F9.2)
5241       RETURN
5261       END
```

TABLE 11 (Cont)

e. Subroutine CATMOD

```
5281        SUBROUTINE CATMOD(IC,I)
5301        COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
5321        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
5341C - - - SQUADRON VARIABLES - - -
5361        COMMON SQNAM(2),NCAT,F(2,10)
5381C - - -CATEGORY VARIABLES - - -
5401        COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
5421        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
5441       &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
5461       &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
5481C - - -
5501        COMMON SI(25),SO(25),SL(25),ATRI(25)
5521        DIMENSION T(2),IN(3)
5541        ALPHA NAME,NPLA,NFUEL,T,IN
5561        DATA T/" ENT","ER: "/
5581        DATA IN/"IP  ","INFO","IC/N"/
5601        N=NAC(IC)
5621        GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130,
5641       &140,150,160,170,180,190,200,210,220,230,240,250),I
5661C
5681     10 PRINT 510,I,T
5701    510 FORMAT(I3,2A4,"NEW CATEGORY NAME"/
5721       &"    (MAX OF 12 CHARACTERS) ")
5741        INPUT 15,(NAME(IC,J),J=1,3) ; 15 FORMAT(3A4)
5761        RETURN
5781     20 PRINT 520,I,T
5801    520 FORMAT(I3,2A4,"NUM. OF TYPES OF AIRCRAFT"/
5821       &"    (MAX OF 3) " )
5841        INPUT,NAC(IC)
5861        RETURN
5881     30 PRINT 530,I,T
5901    530 FORMAT(I3,2A4,"WEEKS TO COMPLETE TRAINING")
5921        INPUT,PHADUR(IC)
5941        RETURN
5961     40 PRINT 540,I,T
5981    540 FORMAT(I3,2A4,"ATTRITION RATE")
6001        INPUT,ATR(IC)
6021        RETURN
```

TABLE 11 (Cont)

e. Subroutine CATMOD (Cont)

```
6041     50 PRINT 550,I,T
6061    550 FORMAT(I3,2A4,"ATTRITION POINT")
6081        INPUT,ATP(IC)
6101        RETURN
6121     60 PRINT 560,I,T
6141    560 FORMAT(I3,2A4,"FLIGHT INSTRUCTOR TOUR OF DUTY(MONTHS)")
6161        PRINT 562
6181    562 FORMAT("    ENTER VALUES FOR IP,INFO,IC/N ")
6201        INPUT,(FITOD(IC,J),J=1,3)
6221        RETURN
6241     70 PRINT 570,I,T
6261    570 FORMAT(I3,2A4,"MONTHS TO TRAIN EACH TYPE FLT INSTR.")
6281        PRINT 562
6301        INPUT,(FITR(IC,J),J=1,3)
6321        RETURN
6341     80 PRINT 580,I,T,N
6361    580 FORMAT(I3,2A4,"NAMES FOR THE",I2," AIRCRAFT TYPE"/
6381       &"    (4 CHARACTERS PER NAME)")
6401        INPUT,(NPLA(IC,J),J=1,N)
6421        RETURN
6441     90 PRINT 590,I,T,N
6461    590 FORMAT(I3,2A4,"THE MO FACTOR FOR THE",I2," AIRCRAFT TYPE"//)
6481        INPUT,(AMO(IC,J),J=1,N)
6501        RETURN
6521    100 PRINT 600,I,T,N
6541    600 FORMAT(I3,2A4,"THE WEATHER FACTOR(100%=1.) FOR"/
6561       &"    THE",I2," AIRCRAFT TYPE ")
6581        INPUT,(WX(IC,J),J=1,N)
6601        RETURN
6621    110 PRINT 610,I,T,N
6641    610 FORMAT(I3,2A4,"FUEL TYPE FOR EACH (",I2,") AIRCRAFT"//)
6661        INPUT,(NFUEL(IC,J),J=1,N)
6681        RETURN
6701    120 PRINT 620,I,T,N
6721    620 FORMAT(I3,2A4,"FUEL CONSUMPTION FOR EACH (",I2,
6741       &") AIRCRAFT"//)
6761        INPUT,(GAS(IC,J),J=1,N)
6781        RETURN
```

TABLE 11 (Cont)

e. Subroutine CATMOD (Cont)

```
6801   130 PRINT 630,I,T,N
6821   630 FORMAT(I3,2A4,"S PER FLIGHT HOUR FOR EACH (",I2,
6841      &") AIRCRAFT"//)
6861       INPUT,(COSTFH(IC,J),J=1,N)
6881       RETURN
6901   140 PRINT 640,I,T,N
6921   640 FORMAT(I3,2A4,"A/C FLT. HRS/DAY FOR EACH (",I2,
6941      &") AIRCRAFT"//)
6961       INPUT,(ACFD(IC,J),J=1,N)
6981       RETURN
7001   150 PRINT 650,I,T,N
7021   650 FORMAT(I3,2A4,"A/C HOURS/STUDENT FOR EACH (",I2,
7041      &") AIRCRAFT"//)
7061       INPUT,(ACHS(IC,J),J=1,N)
7081       RETURN
7101   160 PRINT 660,I,T,N
7121       M=3*N
7141   660 FORMAT(I3,2A4,"INSTRUCTION TYPES FOR EACH (",I2,
7161      &") AIRCRAFT"/4X,"USE 1 FOR IP,2 FOR INFO,3 FOR IC/N",
7181      &", 0 FOR NONE"/4X,"USE 3 INTEGERS PER AIRCRAFT
7201      & (E.G. 1,3,0)"//)
7221       INPUT,(INSTMIX(IC,J),J=1,M)
7241       RETURN
7261   170 PRINT 670,I,T,IN(1),N
7281   670 FORMAT(I3,2A4,1X,A4," UTILIZATION(FLY.DAY) FOR EACH (",
7301      &I2,") AIRCRAFT"//)
7321       INPUT,(FINU(IC,1,J),J=1,N)
7341       RETURN
7361   180 PRINT 670,I,T,IN(2),N
7381       INPUT,(FINU(IC,2,J),J=1,N)
7401       RETURN
7421   190 PRINT 670,I,T,IN(3),N
7441       INPUT,(FINU(IC,3,J),J=1,N)
7461       RETURN
```

TABLE 11 (Cont)

e. Subroutine CATMOD (Cont)

```
7481   200 PRINT 700,I,T,IN(1),N
7501   700 FORMAT(I3,2A4,1X,A4," HOURS/STUDENT FOR EACH (",I2,
7521      &") AIRCRAFT"//)
7541       INPUT,(FINHS(IC,1,J),J=1,N)
7561       RETURN
7581   210 PRINT 700,I,T,IN(2),N
7601       INPUT,(FINHS(IC,2,J),J=1,N)
7621       RETURN
7641   220 PRINT 700,I,T,IN(3),N
7661       INPUT,(FINHS(IC,3,J),J=1,N)
7681       RETURN
7701   230 PRINT 730,I,T,IN(1),N
7721   730 FORMAT(I3,2A4,1X,A4," ACD/LSO/WST RATIO FOR EACH (",I2,
7741      &") AIRCRAFT"//)
7761       INPUT,(RLSO(IC,1,J),J=1,N)
7781       RETURN
7801   240 PRINT 730,I,T,IN(2),N
7821       INPUT,(RLSO(IC,2,J),J=1,N)
7841       RETURN
7861   250 PRINT 730,I,T,IN(3),N
7881       INPUT,(RLSO(IC,3,J),J=1,N)
7901       RETURN
7921       END
```

V. PROGRAM CRAW2

5.1 The purpose of program CRAW2 is to accept the data in question 7, calculate the student input, student output attrites and student load, and print out the student statistics.

5.2 Upon entry to program CRAW2, the data for the student statistic arrays are set to zero. The next question depends on the value of IS(2);

- If IS(2) $\geq$ 2, indicating a return from question 8, question 6 is asked.
- If IS(2) = 1, the name of the squadron and the number of categories are printed. For level of complexity 1, question 6 is asked. For levels 2 and 3, question 5 is asked. Then question 6 is asked and the response is verified.

Next, the program determines if the instruction for indicating no further data and choosing the optional squadron summary should be printed.

5.3 Question 7 is asked, and the computer prints out the type of data the user must enter (i.e., based on this input option). These data are entered and verified. Subroutine STUDENT is then called to calculate the student input and output for each category. After return to CRAW2, the attrites and student load are calculated and the student statistics section of the category summary is printed. Next control is passed to program CRAW3.

SUBROUTINE STUDENT

5.4 The purpose of subroutine STUDENT is to calculate the student input and output based on the data entered and the data input option. The data entered are stored in array SO. This subroutine has four parts corresponding to

the four data input options. Each part solves a different algebraic relationship between the data entered and student output. The correct part is selected from the user's data input option.

5.5 Next the student output and student input are calculated and stored in arrays SO and SI respectively. (Note that array SO contains the data entered upon entry to this subroutine and the student output upon return to CRAW2.)

SUBROUTINE NOYES

5.6 The purpose of subroutine NOYES is to read and validate a no (N), or yes (Y), response from the terminal. If the response is valid, the appropriate nonstandard return is taken. The nonstandard return transfers control to the proper statement in the calling program. Return 1 is taken for a no response, Return 2 is taken for a yes response. If the response is invalid, the user must retype it.

VARIABLES AND COMMON

5.7 The new variables added to the common part of memory in CRAW2 are given in Table 12. Additional variables used in this program are also listed in Table 12.

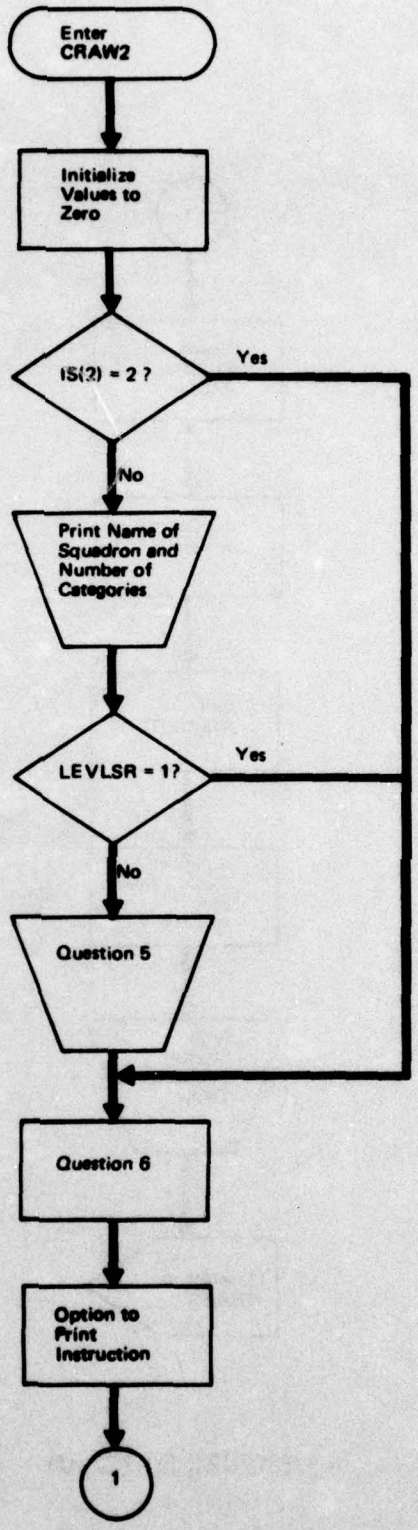


FIGURE 5. PROGRAM CRAW2 FLOW CHART

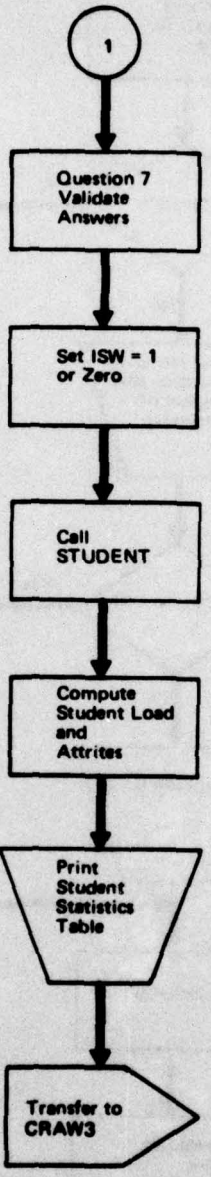


FIGURE 5 (Cont)

a. Subroutine STUDENT

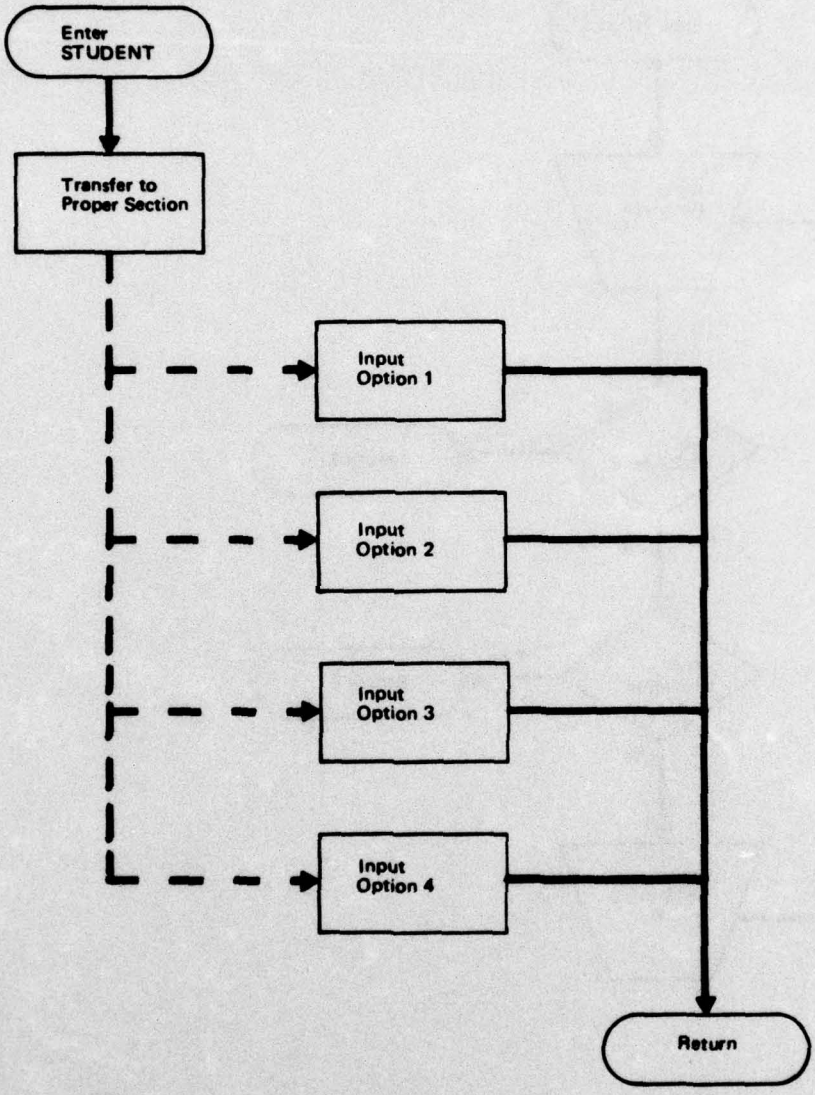


FIGURE 5 (Cont)

b. Subroutine NOYES

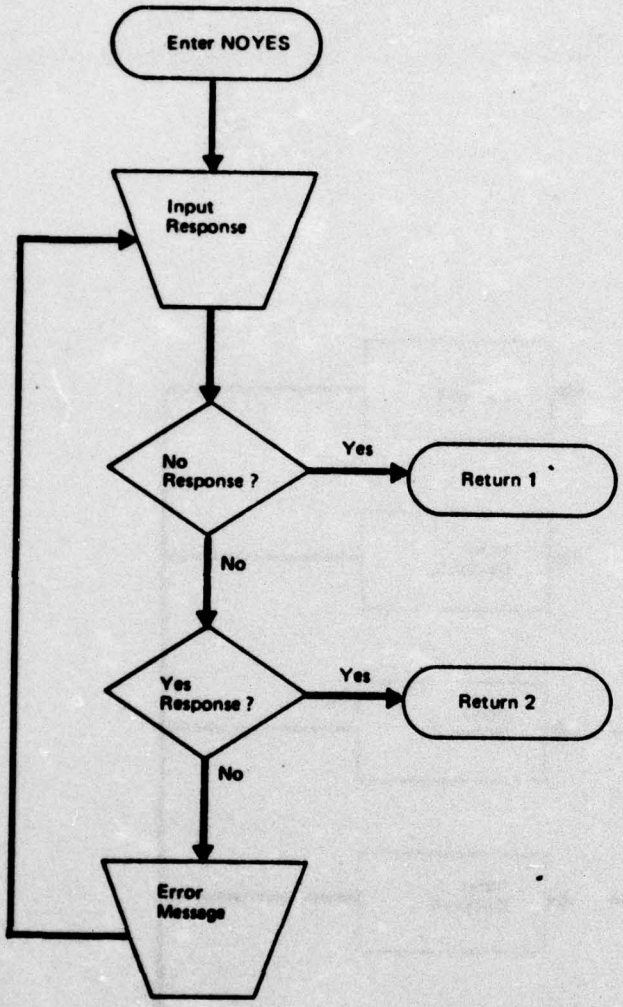


FIGURE 5 (Cont)

TABLE 12

PROGRAM CRAW2 VARIABLE DICTIONARY

Location	Variable Name	Dimension	Description
Common	SI	25	Student input for category I
Common	SO	25	Student output for category I
Common	SL	25	Student load for category I
Common	ATRI	25	Attrites for category I
CRAW2	TITLE	25, 4	Stores the data input option phrases (20 characters per option)
CRAW2	IOP	1	Data input option (between 1 and 4)
CRAW2	N	1	Category number, user input
CRAW2	S	1	Data input value
CRAW2	T1	1	Total attrites in squadron
CRAW2	T2	1	Total student load

TABLE 13

CRAW2 PROGRAM AND SUBROUTINE DICTIONARY

CRAW2	Prints out questions, accepts data for question 7 and validates it. Computes student load and attrites.
STUDENT	Calculates student input and/or output based on data and input option.

TABLE 14

PROGRAM CRAW2 LISTING

```
102C---PROGRAM: CRAW2 (ACCEPTS-CALCULATES STUDS. INPUT)
122       COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
142       COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
162C - - - SQUADRON VARIABLES - - -
182       COMMON SQNAM(2),NCAT,F(2,10)
202C - - -CATAGORY VARIABLES - - -
222       COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9)
242       COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
262      &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
282      &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
302C - - -
322       COMMON SI(25),SO(25),SL(25),ATRI(25)
342       DIMENSION TITLE(5,4)
362       ALPHA NO,YES,TITLE
382       DATA TITLE/"STUD","ENT ","INPU","T TO"," CAT",
402      &"STUD","ENT ","OUTP","UT  "," ",
422      &"THOU","S $ ","FOR ","FLYI","NG",
442      &"NUMB","ER O","F AI","RCRA","FT"/
462       NO="N" ; YES="Y"
482C
502       DO 10 I=1,25
522       SI(I)=0.
542       SO(I)=0.
562       SL(I)=0.
582    10 ATRI(I)=0.
602       IF(IS(2).EQ.2)GO TO 40
622       PRINT 700,SQNAM,NCAT
642       IF(LEVLSR.EQ.1)GO TO 40
662       PRINT 705
682       CALL NOYES($40,$20)
702    20 PRINT 710,SQNAM
722       DO 25 I=1,NCAT
742    25 PRINT 720,I,(NAME(I,J),J=1,3)
762    40 PRINT 740,(I,(TITLE(J,I),J=1,5),I=1,4)
782    43 INPUT,IOP
802       IF( (IOP.GE.1).AND.(IOP.LE.4) )GO TO 50
822       PRINT 790
842       GO TO 43
```

TABLE 14 (Cont)

```
862    50 IF((LEVLSR.EQ.1).OR.(IS(1).GE.2).OR.(IS(2).GT.0))GO TO 60
882       PRINT 745
902    60 PRINT 750,(TITLE(J,IOP),J=1,5)
922C - - NOW ENTER DATA INTO ARRAY SO. TEMPORARY!
942    80 INPUT,N,S
962       IF(N)200,300,100
982   100 IF(N.GT.NCAT)GO TO 200
1002       IF(S.LT.0.)GO TO 200
1022       SO(N)=S
1042       PRINT 760
1062       GO TO 80
1082  200 PRINT 790
1102       GO TO 80
1122C
1142  300 ISW=0
1162       IF(S.GT.0.99)ISW=1
1182       CALL STUDENT(IOP)
1202       T1=0.
1222       T2=0.
1242       PRINT 800,DAT(X),SQNAM
1262       DO 310 I=1,NCAT
1282       ATRI(I)=SI(I)-SO(I)
1302       SL(I)=( SI(I)*ATP(I)+SO(I)*(1.-ATP(I)) )*PHADUR(I)
1322       SL(I)=SL(I)/WPY
1342       T1=T1+ATRI(I)
1362       T2=T2+SL(I)
1382  310 PRINT 810,(NAME(I,J),J=1,3),SI(I),SO(I),ATRI(I),SL(I)
1402       PRINT 820,T1,T2
1422       CHAIN"CRAW3*"
1442C
```

TABLE 14 (Cont)

```
1462   700 FORMAT(" READINESS SQUADRON: ",2A4," HAS ",I2,
1482      & " CATEGORIES" )
1502   705 FORMAT(/" Q-5.  DO YOU WANT A LIST OF CATEGORIES"/
1522      &"     AND THEIR NUMBER(Y,N)")
1542   710 FORMAT(5X,"READINESS SQUADRON: "2A4/
1562      &" CAT. NO.     CAT. NAME")
1582   720 FORMAT(2X,I4,6X,3A4)
1602   740 FORMAT(/" Q-6.  ENTER DATA INPUT OPTION",4(/I4,2X,5A4))
1622   745 FORMAT(/" FOR NO FURTHER DATA ENTER:"/
1642      &" 0,0 TO GET SQUADRON SUMMARY PRINTED"/
1662      &" 0,1 TO SKIP SQUADRON SUMMARY")
1682   750 FORMAT(/" Q-7.  ENTER CAT. NO. AND ",5A4," (XX,XXX)")
1702   760 FORMAT("+NEXT")
1722   790 FORMAT(" INVALID REPLY - RETYPE")
1742   800 FORMAT(//10(4H * *)/1X,A8/5X,"READINESS SQUADRON: ",2A4,
1762      &//12X,"STUDENT STATISTICS"/
1782      &" CAT. NAME",6X,"INPUT  OUTPUT  ATTRITES  LOAD")
1802   810 FORMAT(1X,3A4,4F8.2)
1822   820 FORMAT(" **TOTAL",21X,2F8.2///)
1842       END
```

TABLE 14 (Cont)

a. Subroutine STUDENT

```
1862        SUBROUTINE STUDENT(IOP)
1882        COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
1902        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
1922C - - - SQUADRON VARIABLES - - -
1942        COMMON SQNAM(2),NCAT,F(2,10)
1962C - - -CATAGORY VARIABLES - - -
1982        COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
2002        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
2022       &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
2042       &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
2062        COMMON SI(25),SO(25),SL(25),ATRI(25)
2082C
2102        GO TO (100,200,300,400),IOP
2122C
2142C - - GIVEN STUDENT INPUT (STORED IN ARRAY SO)
2162  100 DO 120 I=1,NCAT
2182        SI(I)=SO(I)
2202  120 SO(I)=SI(I)*(1.-ATR(I))
2222        RETURN
2242C
2262C - -GIVEN STUDENT OUTPUT
2282  200 DO 220 I=1,NCAT
2302  220 SI(I)=SO(I)/(1.-ATR(I))
2322        RETURN
2342C
```

TABLE 14 (Cont)

a. Subroutine STUDENT (Cont)

```
2362C- - GIVEN THOUSAND $ FOR ALL A/C FLYING COSTS
2382   300 DO 320 I=1,NCAT
2402       T=0.
2422       N=NAC(I)
2442       DO 305 J=1,N
2462   305 T=T+COSTFH(I,J)*ACHS(I,J)
2482C - -  T=COST PER STUD OUT
2502       IF(T.LE.0)GO TO 310
2522       SO(I)=SO(I)*1000./T
2542       SI(I)=SO(I)/(1.-ATR(I))
2562       GO TO 320
2582   310 SO(I)=0.
2602       SI(I)=0.
2622   320 CONTINUE
2642       RETURN
2662C
2682C - - SO = NUM. OF A/C OF TYPE ONE
2702   400 DO 420 I=1,NCAT
2722C       FIND STUD OUTPUT/AIRCRAFT
2742       T=ACFD(I,1)*AFD*WX(I,1)/ACHS(I,1)
2762       SO(I)=SO(I)*T
2782   420 SI(I)=SO(I)/(1.-ATR(I))
2802       RETURN
2822       END
```

b. Subroutine NOYES

```
2842      SUBROUTINE NOYES(*,*)
2862       COMMON DUM(12),NO,YES
2882       ALPHA NO,YES,N
2902    10 INPUT 700,N
2922       IF(N.EQ.NO)RETURN1
2942       IF(N.EQ.YES)RETURN2
2962       PRINT," INVALID REPLY - RETYPE"
2982       GO TO 10
3002   700 FORMAT(A1)
3022       END
```

VI. PROGRAM CRAW3

PROGRAM DESCRIPTION

6.1 The purpose of program CRAW3 is to calculate and print the category, aircraft and instructor requirements, and the squadron summary if the user requests it.

6.2 Upon entry the program initializes to zero the variables which contain the total squadron requirements. Next subroutine NAMSORT is called to make a list of aircraft types and associated fuel types for the squadron. Then the program goes through a loop for all categories and calls GENSLR for those categories having a positive student output. In this loop the first argument of GENLSR, IX, is 1 and the aircraft statistics are calculated and printed. In the second loop IX is 2 and the instructor requirements are calculated and printed.

6.3 Subroutine SQDSUM is called if ISW $\neq$ 1 and squadron summary is printed. If ISW = 1, the call to SQDSUM is skipped. Finally question 8 is asked and the program checks the answer and transfers control to the proper program along with the appropriate value in IS(1) and IS(2).

SUBROUTINE NAMSORT

6.4 When this program is called, a list is made of all the different aircraft types (SNPLA) in the squadron along with the proper fuel type (SNFUEL) for the aircraft type. First the aircraft type and the fuel type from the first category are placed in the arrays SNPLA and SNFUEL. NACT is set equal to the number of aircraft types in the category. Then the aircraft types and their corresponding fuel types in each category are compared with those in this array. If there is a match, the aircraft is already in the array. If there is no match, the new aircraft type is added to the list. The total number of aircraft types is checked to be sure it does not exceed three.

6.5 This procedure continues until all categories have been examined. An error message is printed and the program stops if a fourth aircraft type is found or a fuel type for an aircraft does not match.

SUBROUTINE GENLSR

6.6 The purpose of this subroutine is to calculate and print aircraft and instructor requirements. This subroutine has two arguments IX and ICAT. Variable IX refers to the part of the subroutine to be executed (IX = 1 is for aircraft calculations, IX = 2 is for instructor calculations). The variable ICAT refers to the category number.

6.7 If IX = 1, then the program goes to the aircraft section. First, the variables are initialized to zero. Then the flight hours, number of required aircraft, fuel, and enlisted men are calculated as a function of student output. Next the values are added to the arrays for the squadron total. Finally the results for the category are printed and control returns to the main program.

6.8 If IX = 2 when the subroutine is entered, then the instructor calculation section is executed. First, the appropriate instructor category values are initialized to zero. Then, based on the student output and the instruction mix, the required number of instructor pilots (IP), instructors under training (IUT), and others (ACD/LSO/WST) are calculated. Then the values are added to the arrays for the squadron total. Finally the results for the category are printed and control returns to the main program.

SUBROUTINE SQDSUM

6.9 The purpose of SQDSUM is to calculate and print the squadron summary.

VARIABLES AND COMMON

6.10 More variables are added to common as described in Table 15. Also additional variables are used in GENLSR.

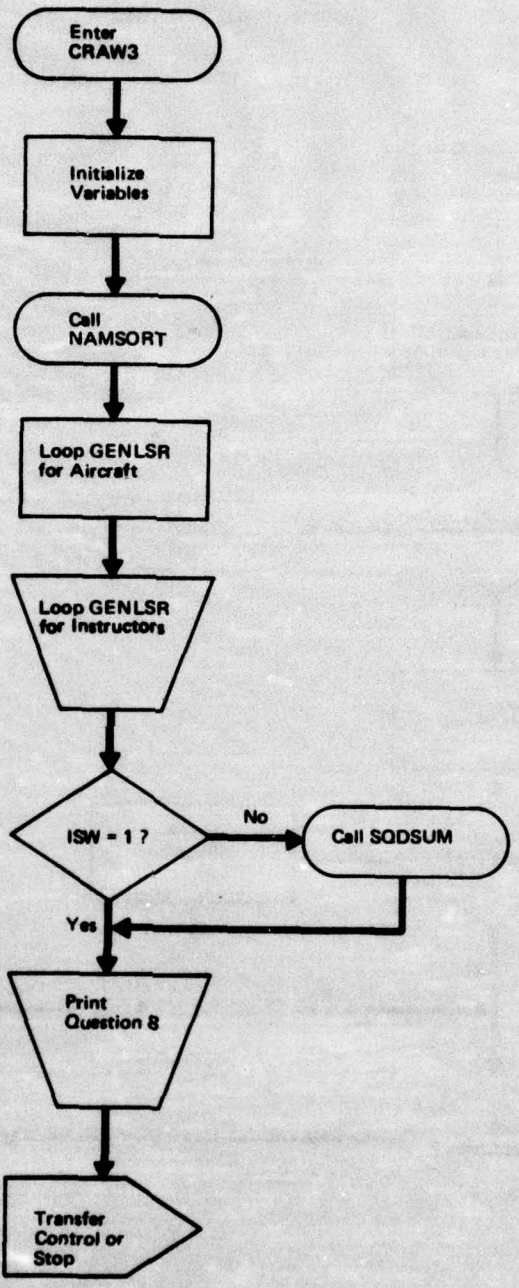


FIGURE 6. PROGRAM CRAW3 FLOW CHART

a. Subroutine NAMSORT

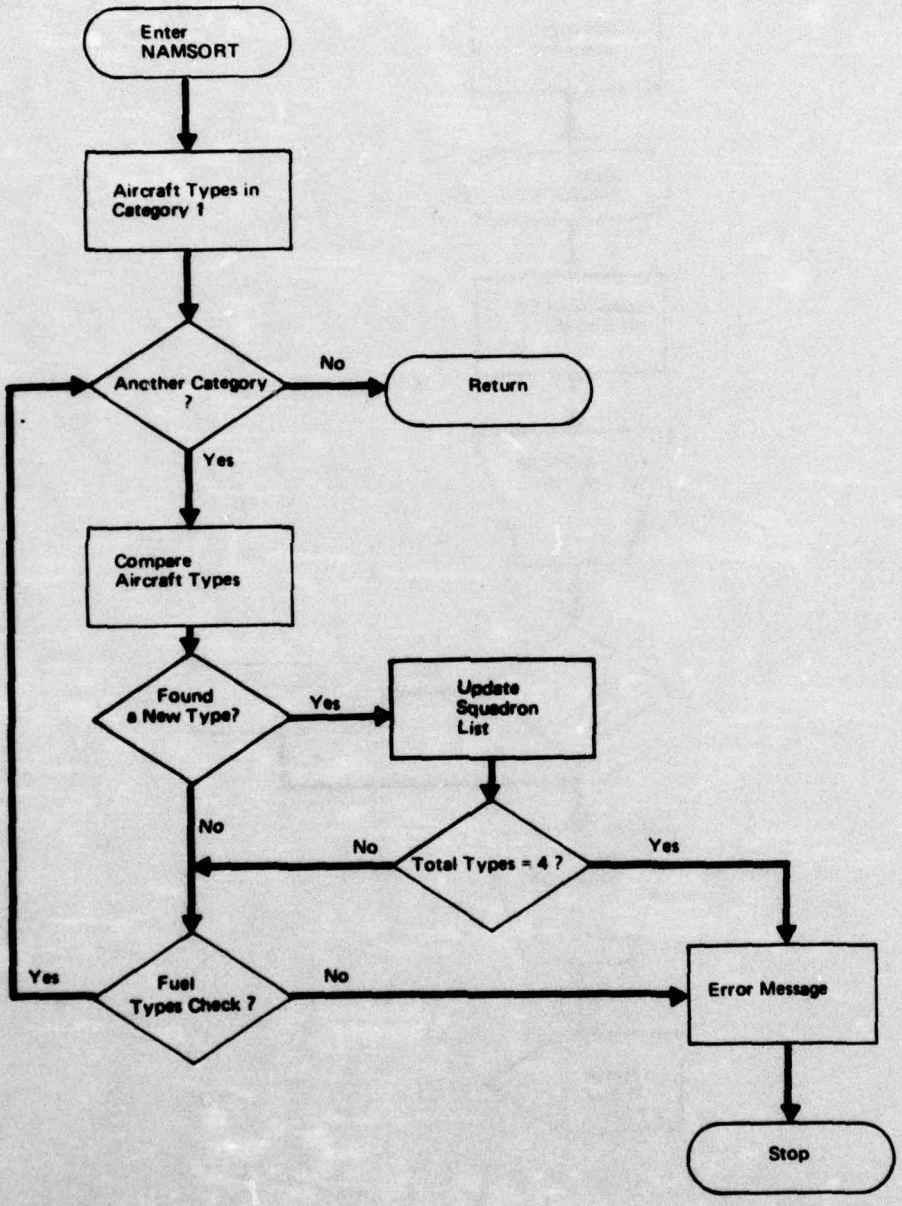


FIGURE 6 (Cont)

b. Subroutine GENLSR

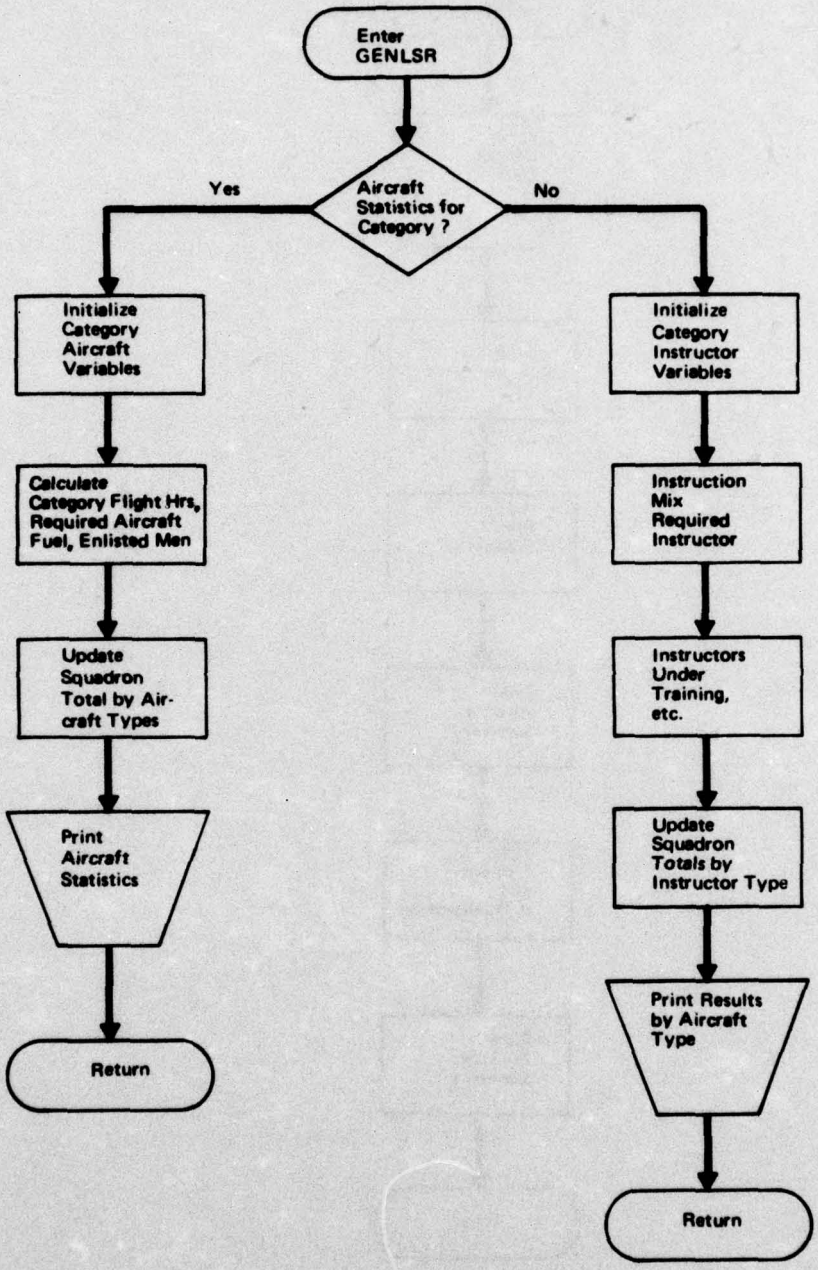


FIGURE 6 (Cont)

c. Subroutine SQDSUM

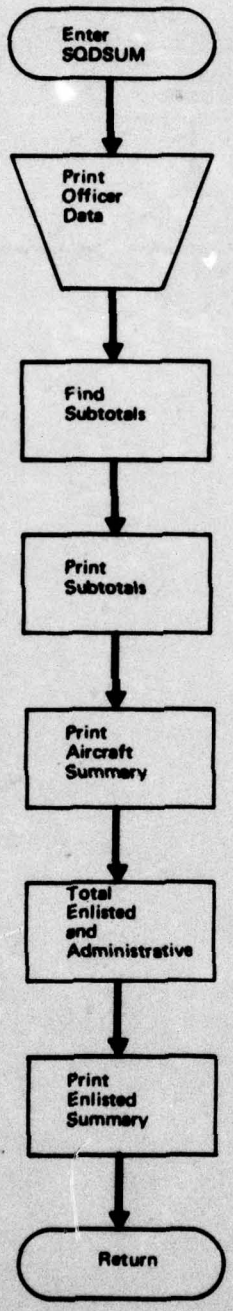


FIGURE 6 (Cont)

TABLE 15

PROGRAM CRAW3 VARIABLE DICTIONARY

Location	Name	Dimension	Description
Common	SNPLA	3	Types of different aircraft in squadron
Common	SNFUEL	3	Fuel associated with aircraft type I in squadron
Common	NACT	1	Total number of different aircraft types in squadron
Common	TAC	3	Total aircraft type I required for squadron
Common	TFH	3	Total squadron flight hours for aircraft type I
Common	TCOST	3	Total squadron cost for aircraft type I
Common	TGAS	3	Total fuel for squadron for aircraft type I
Common	TFIN	3	Total squadron flight instructors for aircraft type I
Common	TIUT	3	Total squadron instructors under training for aircraft type I
Common	TLSO	3	Total ACD/LSO/WST instructors for squadron for aircraft type I
Common	TEM	3	Total enlisted maintenance men for squadron for aircraft type I
Common	DUM	14	First 14 locations of common
GENLSR	CAC	3	Total required aircraft of type I in category
GENLSR	CFH	3	Category flight hours for aircraft type I
GENLSR	CCOST	3	Category flight cost for aircraft type I
GENLSR	CGAS	3	Category fuel for aircraft type I

TABLE 15 (Cont)

Location	Name	Dimension	Description
GENLSR	CINS	3,3	Category instructors aircraft type I, instruction type J
GENLSR	CIUT	3,3	Category instructors under training aircraft type I, instructor type J
GENLSR	CLSO	3,3	Category ACD/LSO/WST instructors for aircraft type I, instructor type J
GENLSR	CEM	3	Enlisted men required for aircraft type I for the category
SQDSUM	TO	1	Total enlisted men
SQDSUM	FACT	1	Factor for enlisted administrative
SQDSUM	A	1	Number of administrative personnel for enlisted

TABLE 16

CRAW3 PROGRAM AND SUBROUTINE DICTIONARY

CRAW3	Main program to call other subroutines and transfer control to appropriate program
NAMSORT	Makes list of different aircraft types in the squadron
GENLSR	Calculate and print aircraft and instructor requirements for a category
SQDSUM	Prints squadron summary

TABLE 17

PROGRAM CRAW3 LISTING

```
103C---PROGRAM: CRAW3  (CALCULATES A/C & INST. ETC)
123       COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
143       COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
163C - - - SQUADRON VARIABLES - - -
183       COMMON SQNAM(2),NCAT,F(2,10)
203C - - -CATAGORY VARIABLES - - -
223       COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9)
243       COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
263      &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
283      &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
303C - - -
323       COMMON SI(25),SO(25),SL(25),ATRI(25)
343       COMMON SNPLA(3),SNFUEL(3),NACT,
363      & TAC(3),TFH(3),TCOST(3),TGAS(3),
383      & TFIN(3),TIUT(3),TLSO(3),TEM(3)
403C
423       DO 10 I=1,3
443       TAC(I)=0.
463       TFH(I)=0.
483       TCOST(I)=0.
503       TGAS(I)=0.
523       TFIN(I)=0.
543       TIUT(I)=0.
563       TEM(I)=0.
583    10 TLSO(I)=0.
603       CALL NAMSORT
623C
643C - - CATAGORY SUMMARY
663       PRINT 600
683       PRINT 610
703       DO 30 I=1,NCAT
723       IF(SO(I).LE.0.)GO TO 30
743       CALL GENLSR(1,I)
763    30 CONTINUE
783       PRINT 620
803       PRINT 630
823       DO 40 I=1,NCAT
843       IF(SO(I).LE.0.)GO TO 40
863       CALL GENLSR(2,I)
883    40 CONTINUE
```

TABLE 17 (Cont)

```
903C - - SQUADRON SUMMARY
923        IF(ISW.EQ.1)GO TO 50
943        PRINT 650
963        CALL SQDSUM
983     50 PRINT 660
1003C
1023        IF(LEVLSR.LE.2)PRINT 700
1043        IF(LEVLSR.EQ.3)PRINT 710
1063     95 INPUT,ISW
1083        IS(2)=0
1103        IF(ISW.EQ.0)GO TO 200
1123        IF(ISW.EQ.1)CHAIN"CRAWM*"
1143        IS(2)=1
1163        IF(ISW.EQ.4)CHAIN"CRAWH*"
1183        IS(2)=2
1203        IF(ISW.EQ.6)CHAIN"CRAW2*"
1223        IF((LEVLSR.EQ.3).AND.(ISW.EQ.21))CHAIN"CRAW1*"
1243        PRINT,"INVALID REPLY - RETYPE"
1263        GO TO 95
1283    200 PRINT 740
1303        STOP
1323C
1343    600 FORMAT(12X"AIRCRAFT STATISTICS")
1363    610 FORMAT(" CAT. NAME",4X,"TYPE    NUM.  FLT.HRS.",
1383       &4X,"COST  GALLONS  FUEL    MO"/
1403       &26X,"- - - - - (X1000)- - - - -")
1423    620 FORMAT(//12X"INSTRUCTOR STATISTICS")
1443    630 FORMAT(" CAT. NAME",4X,"A/C ",
1463       &"*  INSTRUCTORS    ** UNDER TRAINING *",
1483       &"*  ACD/LSO/WST    *"/
1503       &14X,"TYPE",3("  IP    INFO  IC/N") )
1523    650 FORMAT(//"   SQUADRON SUMMARY"//)
1543    660 FORMAT(//10(4H * *))
1563
1583    700 FORMAT(//" Q-8. RETURN TO QUESTION 1,4, OR 6."/
1603       &" ENTER 0 TO STOP (X)")
1623    710 FORMAT(//" Q-8. RETURN TO QUESTION 1,4,6 OR 21"/
1643       &" ENTER 0 TO STOP (X)")
1663    740 FORMAT(//10(4H - -)///)
1683        END
```

TABLE 17 (Cont)

a. Subroutine NAMSORT

```
1703        SUBROUTINE NAMSORT
1723        COMMON DUM(14)
1743        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
1763C - - - SQUADRON VARIABLES - - -
1783        COMMON SQNAM(2),NCAT,F(2,10)
1803C - - -CATAGORY VARIABLES - - -
1823        COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
1843        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
1863       &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
1883       &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
1903C - - -
1923        COMMON SI(25),SO(25),SL(25),ATRI(25)
1943        COMMON SNPLA(3),SNFUEL(3),NACT
1963        ALPHA SNPLA,NPLA,SNFUEL,NFUEL
1983C
2003        NACT=NAC(1)
2023        DO 10 I=1,NACT
2043        SNFUEL(I)=NFUEL(1,I)
2063     10 SNPLA(I)=NPLA(1,I)
2083        IF(NCAT.EQ.1)RETURN
2103C
2123        DO 50 I=2,NCAT
2143        N=NAC(I)
2163        DO 30 J=1,N
2183        DO 20 K=1,NACT
2203        IF(SNPLA(K).EQ.NPLA(I,J))GO TO 25
2223     20 CONTINUE
2243C - - FOUND A NEW AIRCRAFT TYPE
2263        NACT=NACT+1
2283        IF(NACT.GT.3)GO TO 500
2303        SNPLA(NACT)=NPLA(I,J)
2323        SNFUEL(NACT)=NFUEL(I,J)
2343        GO TO 30
2363     25 IF(SNFUEL(K).NE.NFUEL(I,J))GO TO 600
2383     30 CONTINUE
2403     50 CONTINUE
2423        RETURN
```

TABLE 17 (Cont)

a. Subroutine NAMSORT (Cont)

```
2443C
2463   500 PRINT 510,(NAME(I,L),L=1,3),NPLA(I,J)
2483       PRINT 620
2503       STOP
2523   510 FORMAT(//" ERROR IN: ",3A4/
2543      &"  A FOURTH AIRCRAFT TYPE FOR THE SQUADRON: ",A4)
2563   600 PRINT 610,(NAME(I,L),L=1,3),NFUEL(I,J),NPLA(I,J)
2583       PRINT 620
2603       STOP
2623   610 FORMAT(//" ERROR IN: ",3A4/
2643      &"  FUEL TYPE ",A4," FOR AIRCRAFT ",A4/
2663      &"  DOES NOT MATCH FUEL ASSIGNMENTS FOR THIS"/
2683      &"  AIRCRAFT IN PREVIOUS CATEGORIES")
2703   620 FORMAT(//" * CORRECT DATA FILE AND RERUN"///)
2723       END
```

TABLE 17 (Cont)

b. Subroutine GENLSR

```
2743        SUBROUTINE GENLSR(IX,ICAT)
2763        COMMON DUM(14)
2783        COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
2803C - - - SQUADRON VARIABLES - - -
2823        COMMON SQNAM(2),NCAT,F(2,10)
2843C - - -CATAGORY VARIABLES - - -
2863        COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
2883        COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
2903       &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
2923       &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
2943C - - -
2963        COMMON SI(25),SO(25),SL(25),ATRI(25)
2983        COMMON SNPLA(3),SNFUEL(3),NACT,
3003       & TAC(3),TFH(3),TCOST(3),TGAS(3),
3023       & TFIN(3),TIUT(3),TLSO(3),TEM(3)
3043C
3063        DIMENSION CAC(3),CFH(3),CCOST(3),CGAS(3),
3083       & CINST(3,3),CIUT(3,3),CLSO(3,3),CEM(3)
3103        ALPHA SNPLA,NPLA,SNFUEL,NFUEL
3123C
3143        N=NAC(ICAT)
3163        GO TO (10,200),IX
3183C
3203C - - AIRCRAFT STATISTICS FOR CAT
3223     10 DO 15 I=13
3243        CAC(I)=0.
3263        CFH(I)=0.
3283        CCOST(I)=0.
3303        CEM(I)=0.
3323     15 CGAS(I)=0.
3343        DO 30 I=1,N
3363        CFH(I)=SO(ICAT)*ACHS(ICAT,I)
3383        T=ACFD(ICAT,I)*AFD*WX(ICAT,I)
3403        IF(T.LE.0.)GO TO 20
3423        CAC(I)=CFH(I)/T
3443     20 CCOST(I)=COSTFH(ICAT,I)*CFH(I)/1000.
3463        CGAS(I)=GAS(ICAT,I)*ACHS(ICAT,I)/1000.
3483        CGAS(I)=CGAS(I)*SO(ICAT)
3503        CEM(I)=AMO(ICAT,I)*CAC(I)
3523     30 CFH(I)=CFH(I)/1000.
```

TABLE 17 (Cont)

b. Subroutine GENLSR (Cont)

```
3543C - - UPDATE SQUADRON TOTAL
3563      DO 50 I=1,N
3583      DO 40 K=1,NACT
3603      IF(SNPLA(K).EQ.NPLA(ICAT,I))GO TO 45
3623   40 CONTINUE
3643      PRINT," MACHINE ERROR IN CRAW3/GENLSR - RERUN PROGRAM"
3663      STOP
3683   45 TAC(K)=TAC(K)+CAC(I)
3703      TFH(K)=TFH(K)+CFH(I)
3723      TCOST(K)=TCOST(K)+CCOST(I)
3743      TGAS(K)=TGAS(K)+CGAS(I)
3763      TEM(K)=TEM(K)+CEM(I)
3783   50 CONTINUE
3803      PRINT 600,(NAME(ICAT,J),J=1,3),NPLA(ICAT,1),CAC(1),
3823     & CFH(1),CCOST(1),CGAS(1),NFUEL(ICAT,1),AMO(ICAT,1)
3843      IF(N.EQ.1)GO TO 95
3863      DO 60 I=2,N
3883   60 PRINT 610,NPLA(ICAT,I),CAC(I),CFH(I),
3903     & CCOST(I),CGAS(I),NFUEL(ICAT,I),AMO(ICAT,I)
3923   95 RETURN
3943C
3963C - - INSTRUCTOR STATISTICS
3983  200 DO 210 I=1,3
4003      DO 210 J=1,3
4023      CINST(I,J)=0.
4043      CIUT(I,J)=0.
4063  210 CLSO(I,J)=0.
4083C - - COMPUTE FLT. INST. BY A/C
4103      K=0
4123      DO 240 I=1,N
4143      DO 230 J=1,3
4163      K=3*(I-1)+J
4183      M=INSTMIX(ICAT,K)
4203      IF( (M.LE.0).OR.(M.GT.3) )GO TO 230
4223C       THUS: I=A/C TYPE , M=INSTRUCTOR TYPE
4243      T=FINU(ICAT,M,I)*WX(ICAT,I)*AFD
4263      IF(T.LE.0.)GO TO 230
4283      CINST(I,M)=SO(ICAT)*FINHS(ICAT,M,I)/T
4303  230 CONTINUE
4323  240 CONTINUE
4343C
```

TABLE 17 (Cont)

b. Subroutine GENLSR (Cont)

```
4363        DO 244 I=1,N
4383        T=FITOD(ICAT,I)
4403        IF(T.GT.0.)GO TO 241
4423        T=0.
4443        GO TO 242
4463   241  T=FITR(ICAT,I)/T
4483   242  DO 244 M=1,3
4503        CIUT(I,M)=T*CINST(I,M)
4523        X=RLSO(ICAT,M,I)
4543        IF(X.LE.0.) GO TO 244
4563        CLSO(I,M)=SL(ICAT)/X
4583   244  CONTINUE
4603C  - -  UPDATE SQUADRON TOTAL
4623        DO 280 M=1,3
4643        DO 280 I=1,3
4663        TFIN(M)=TFIN(M)+CINST(I,M)
4683        TIUT(M)=TIUT(M)+CIUT(I,M)
4703        TLSO(M)=TLSO(M)+CLSO(I,M)
4723   280  CONTINUE
4743        PRINT 700,(NAME(ICAT,J),J=1,3),NPLA(ICAT,1),
4763       &(CINST(1,J),J=1,3),(CIUT(1,J),J=1,3),
4783       &(CLSO(1,J),J=1,3)
4803        IF(N.EQ.1) GO TO 295
4823        DO 285 I=2,N
4843   285  PRINT 710,NPLA(ICAT,I),(CINST(I,J),J=1,3),
4863       & (CIUT(I,J),J=1,3),(CLSO(I,J),J=1,3)
4883   295  RETURN
4903C
4923   600  FORMAT(1X,3A4,1X,A4,4(1X,F8.2),1X,A4,1X,F7.2)
4943   610  FORMAT(14X,A4,4(1X,F8.2),1X,A4,1X,F7.2)
4963   700  FORMAT(1X,3A4,1X,A4,9F6.2)
4983   710  FORMAT(14X,A4,9F6.2)
5003        END
```

TABLE 17 (Cont)

c. Subroutine SQDSUM

```
      SUBROUTINE SQDSUM
      COMMON IY,ISW,LEVLSR,IS(7),KILL,IBC,NO,YES
      COMMON SW(2),AFD,WPY,ALLSQD(30,2),NFILE1,NSQD
C - - - SQUADRON VARIABLES - - -
      COMMON SQNAM(2),NCAT,F(2,10)
C - - -CATAGORY VARIABLES - - -
      COMMON  NAME(25,3),NPLA(25,3),NFUEL(25,3),NAC(25),INSTMIX(25,9
      COMMON AMO(25,3),WX(25,3),PHADUR(25),ATR(25),ATP(25),
     &GAS(25,3),COSTFH(25,3),FITOD(25,3),FITR(25,3),
     &ACFD(25,3),ACHS(25,3),FINU(25,3,3),FINHS(25,3,3),RLSO(25,3,3)
C - - -
      COMMON SI(25),SO(25),SL(25),ATRI(25)
      COMMON SNPLA(3),SNFUEL(3),NACT,
     & TAC(3),TFH(3),TCOST(3),TGAS(3),
     & TFIN(3),TIUT(3),TLSO(3),TEM(3)
      DIMENSION T(3)
      PRINT,"OFFICERS"
      PRINT 700
      PRINT 710,(TFIN(I),I=1,3)
      PRINT 720,(TIUT(I),I=1,3),(TLSO(I),I=1,3)
      PRINT 750,F(1,1),F(1,2)
C - - NOW GET SUBTOTAL
      T(1)=TFIN(1)+TIUT(1)+TLSO(1)+F(1,1)
      T(2)=TFIN(2)+TIUT(2)+TLSO(2)+F(1,2)
      T(3)=TFIN(3)+TIUT(3)+TLSO(3)
      PRINT 760,(T(I),I=1,3)
      TO=T(1)+T(2)+T(3)
      TO=TO+F(1,3)+F(1,4)+F(1,5)
      PRINT 770,F(1,3),F(1,4),F(1,5),TO
C
```

TABLE 17 (Cont)

c. Subroutine SQDSUM (Cont)

```
5623C - - NOW GIVE A/C SUMMARY
5643      PRINT,"AIRCRAFT"
5663      PRINT 800,(SNPLA(I),I=1,NACT)
5683      PRINT 810,(TAC(I),I=1,NACT)
5703      PRINT 813,(TFH(I),I=1,NACT)
5723      PRINT 815,(TCOST(I),I=1,NACT)
5743      PRINT 820,(SNFUEL(I),I=1,NACT)
5763      PRINT 825,(TGAS(I),I=1,NACT)
5783      PRINT 830,(TEM(I),I=1,NACT)
5803C
5823C - - NOW GIVE ENLISTED SUPPORT
5843      PRINT 900
5863      TO=TEM(1)+TEM(2)+TEM(3)
5883      FACT=1.2
5903      IF(TO-200.)70,50,40
5923   40 IF(TO-400.)50,60,60
5943   50 FACT=1.15
5963      GO TO 70
5983   60 FACT=1.10
6003   70 A=TO*(FACT-1.)
6023      PRINT 910, TO,A
6043      PRINT 920,(F(2,I),I=1,5)
6063      TO=TO+A
6083      DO 80 I=1,5
6103   80 TO=TO+F(2,I)
6123      PRINT 930,TO
6143C
```

TABLE 17 (Cont)

c. Subroutine SQDSUM (Cont)

```
6163   700 FORMAT("   AVIATORS",8X,"IP",5X,"INFO",4X,"IC/N")
6183   710 FORMAT(5X,"INST.",5X,4F8.2)
6203   720 FORMAT(5X,"IUT",7X,3F8.2/5X,"ACD/LSO/WST",F7.2,2F8.2)
6223   750 FORMAT(5X,"ADMIN."4X,2F8.2)
6243   760 FORMAT(15X,3(8H  ------)/" **SUBTOTAL",4X,4F8.2)
6263   770 FORMAT("   NON-AVIATORS"/4X,"GRD.ADMIN.",F8.2/
6283      &4X,"MAINT.GRD.",F8.2/
6303      &4X,"OTHER     ",F8.2/
6323      &" **TOTAL OFFICERS: ",F8.2//)
6343C
6363   800 FORMAT(4X,"TYPE",8X,3(A4,4X) )
6383   810 FORMAT(4X,"NUM.",4X,3F8.2)
6403   813 FORMAT(2X,"HRS.(1000)",3F8.2)
6423   815 FORMAT(2X,"COST(1000)",3F8.2)
6443   820 FORMAT(4X,"FUEL",8X,3(A4,4X))
6463   825 FORMAT(2X,"GALS(1000)",3F8.2)
6483   830 FORMAT(1X,"MAIN.ENL.",2X,3F8.2)
6503C
6523   900 FORMAT(//" ENLISTED")
6543   910 FORMAT(4X,"MAINT.",5X,F7.2/4X,"ADMIN.",5X,7.2)
6563   920 FORMAT(4X,"TRNG.SUPP. ",F7.2/
6583      & 4X,"DET. SUPP. ",F7.2/
6603      & 4X,"SITE SUPP. ",F7.2/
6623      & 4X,"ADMN.SUPP. ",F7.2/
6643      & 4X,"CREWS",6X,F7.2)
6663   930 FORMAT(" ***TOTAL",5X,F8.2)
6683       RETURN ; END
```

VII. PROGRAM XHUNT

PROGRAM DESCRIPTION

7.1 The purpose of program XHUNT is to read and print various lines from a data file that has a structure similar to that of SQUAD*1 or SQUAD*2. Program XHUNT asks the user the appropriate questions, validates the user response, opens the data file, then calls subroutine SUB. Upon returning from the subroutine by a nonstandard return, the data file is closed and the program stops.

SUBROUTINE SUB

7.2 The purpose of subroutine SUB is to read the data file, and print out the appropriate line numbers. The arguments for the subroutine are I and L where I is the print option and L is the category data line reference number. A nonstandard return is also used.

7.4 If I = 1, the first line of data for each squadron is read and printed until the name is "END␣" (i.e., the line number, squadron name, and number of categories (N) are read and printed). The name is then checked for "END␣" indicating an end of data file. If it is not the end of the data file, then the next 2 + 20N data lines are skipped (i.e., 2 for next 2 lines of squadron data plus 20 lines for each category.) The lines must be read, but nothing is done with the data.

7.5 If I = 2 or 3, then a category data line is printed. In this case after the squadron name is read, the next two lines of squadron data are skipped. Then indices of the required data are set up. The appropriate number of category data lines are skipped, the required line is read under an alpha format into the array C. Thus, only a maximum of the first 40 characters, including the line

number, can be read and printed. Next the subroutine determines how many trailing words in C are blank and they are not printed. Then the remaining data lines in the category are skipped.

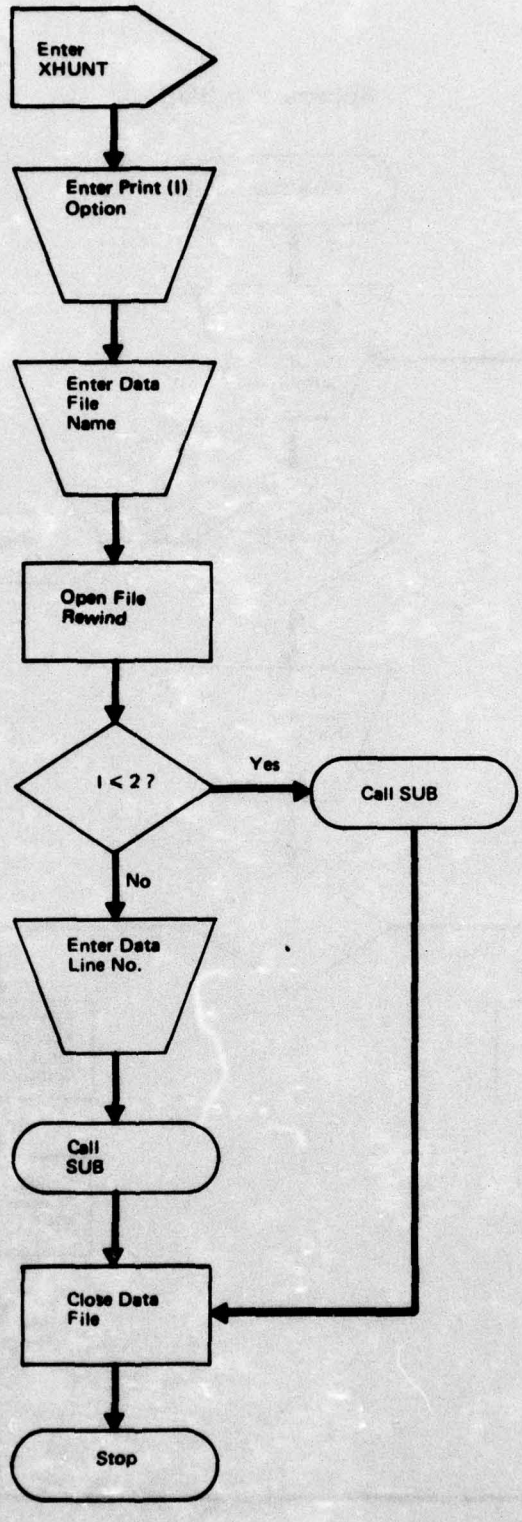


FIGURE 7. PROGRAM XHUNT FLOW CHART

Subroutine SUB

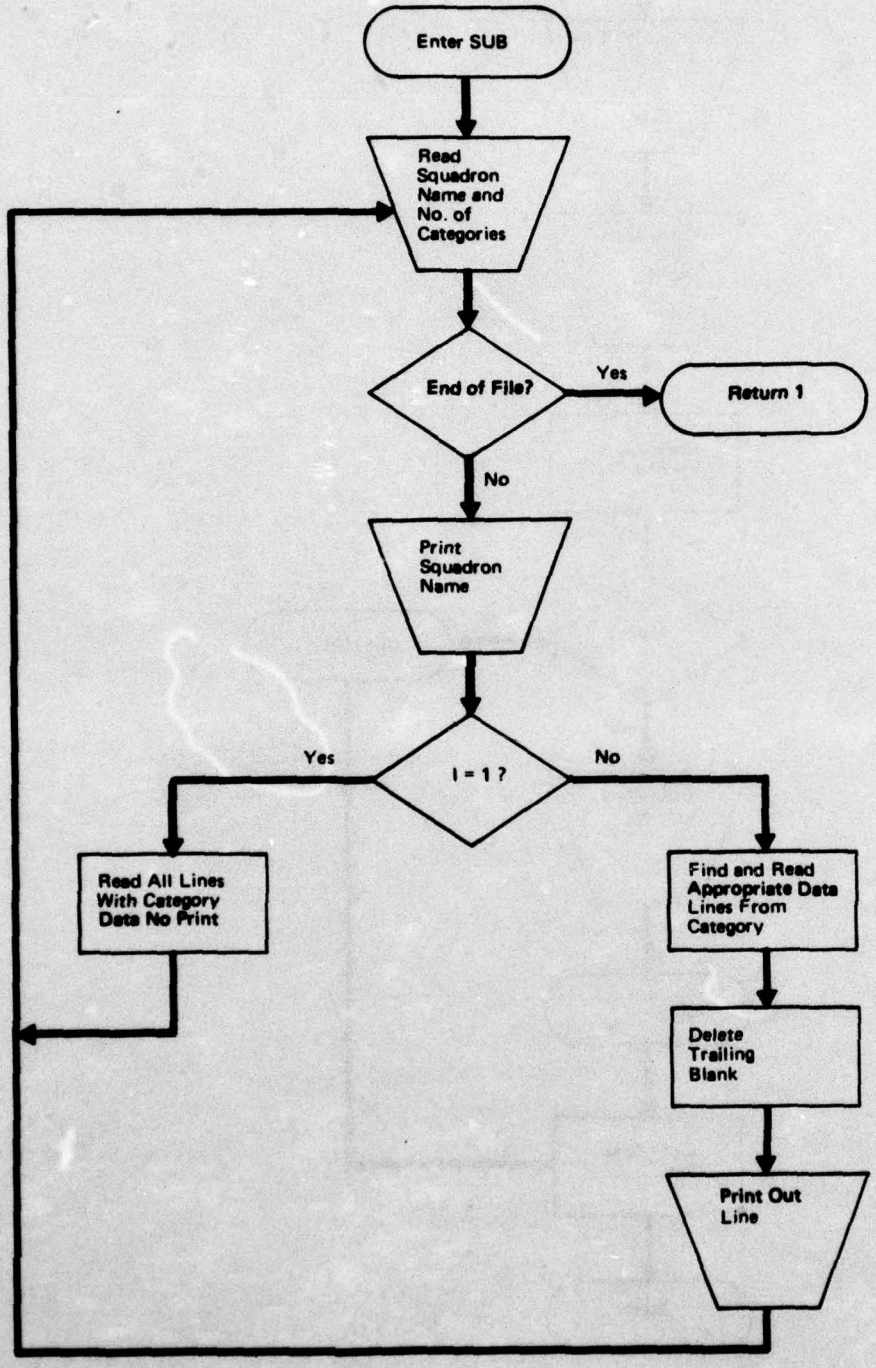


FIGURE 7 (Cont)

TABLE 18

PROGRAM XHUNT VARIABLE DICTIONARY

Variable Name	Description
T	File name
EOF	Contains characters "ENDb"
C	Array into which data are read
I	Print option between 1 and 3
L	Category data line number between 1 and 20
IL	Data file line number
NAM1	First 4 characters of SQUADRON name
NAM2	Last 4 characters of SQUADRON name
N	Number of categories in squadron
NI	Number of lines to be skipped before required category data
N2	Number of lines to be skipped to get to next category name
K2	Number of largest nonblank word in array C

TABLE 19

XHUNT PROGRAM AND SUBROUTINE DICTIONARY

XHUNT	Asks questions, accepts input, opens and closes data file
SUB	Reads the data file and prints the data

TABLE 20

PROGRAM XHUNT LISTING

```
100C---PROGRAM: XHUNT  (READ & PRINT DATA FILE LINES)
110C            FOR FILES SQUAD*1 AND SQUAD*2 OR COPIES
120       COMMON T,EOF,C(10)
130       ALPHA EOF,C
140       FILENAME T
150       EOF="END "
160       PRINT 800
170    10 INPUT,I
180       IF((I.GE.1).AND.(I.LE.3))GO TO 20
190       PRINT 810
200       GO TO 10
210C
220    20 PRINT 820
230       INPUT,T
240       OPENFILE T
250       REWIND T
260       IF(I.EQ.1) CALL SUB($100,I,0)
270       IF(I.EQ.2) CALL SUB($100,I,1)
280       PRINT 830
290    30 INPUT,L
300       IF((L.GE.1).AND.(L.LE.20))GO TO 40
310       PRINT 810
320       GO TO 30
330    40 CALL SUB($100,I,L)
340   100 CLOSEFILE T
350       PRINT 850
360C
370   800 FORMAT(" ENTER PRINT OPTION:"/
380      &"    1 SQUAD NAME ONLY"/
390      &"    2 SQUAD AND CAT NAME"/
400      &"    3 SQUAD NAME AND CAT DATA LINE NO. ")
410   810 FORMAT(" INVALID REPLY - RETYPE")
420   820 FORMAT(" ENTER FILE NAME ")
430   830 FORMAT(" ENTER DATA LINE NO. ")
440   850 FORMAT(//)
450       STOP;END
```

TABLE 20 (Cont)

Subroutine SUB

```
      SUBROUTINE SUB(*,I,L)
      COMMON T,EOF,C(10)
      ALPHA EOF,C,NAM1,NAM2
      FILENAME T
C
   10 IF(I.GE.2)PRINT," "
      READ(T,700)IL,NAM1,NAM2,N
      PRINT 702,IL,NAM1,NAM2,N
      IF(NAM1.EQ.EOF)GO TO 100
      GO TO (20,50,50),I
C - - - - SQUAD NAME ONLY
   20 M=2+20*N
      DO 25 J=1,M
   25 READ(T,701)IL
      GO TO 10
C - - - - OTHER OPTION
   50 READ(T,701)IL
      READ(T,701)IL
      N1=L-1
      N2=20-L
C - - - - CYCLE FOR EACH CAT
      IF(N.LE.0)GO TO 10
      DO 80 I2=1,N
      IF(N1.EQ.0)GO TO 60
      DO 55 J=1,N1
   55 READ(T,701)IL
   60 READ(T,705)C
C - - - DELETE TRAILING BLANKS IN C ARRAY
      DO 65 K1=1,10
      K2=11-K1
      IF(C(K2).NE."    ")GO TO 70
   65 CONTINUE
   70 PRINT 706,(C(K),K=1,K2)
      IF(N2.EQ.0)GO TO 80
      DO 75 J=1,N2
   75 READ(T,701)IL
   80 CONTINUE
      GO TO 10
  100 RETURN1
C
  700 FORMAT(I4,1X,2A4,I4)
  701 FORMAT(V)
  702 FORMAT(1X,I4,1X,2A4,I4)
  705 FORMAT(10A4)
  706 FORMAT(1X,10A4)
      END
```